

Abril 1985·250 ptas.

TodoSpectrum

AÑO 1 - NUMERO 8.

REVISTA EXCLUSIVA PARA USUARIOS

Amplía tu memoria... a 48K

Aula informática con Spectrum

Especial arquitectos: PREYME

Fútbol americano, manager

Listado del monopoly





Portada: DINAMIC

DIRECTOR: Simeón Cruz
COORDINADOR EDITORIAL: Emiliano Juárez
REDACCION: Juan Arencibia, Fernando García, José C. Tomás, Gumersindo García, Luis M. Brugarolas, Ricardo García, Santiago Gala
DISEÑO: Ricardo Segura

Editado por PUBLINFORMATICA, S. A.
Presidente: Fernando Bolín
Director Editorial: Norberto Gallego

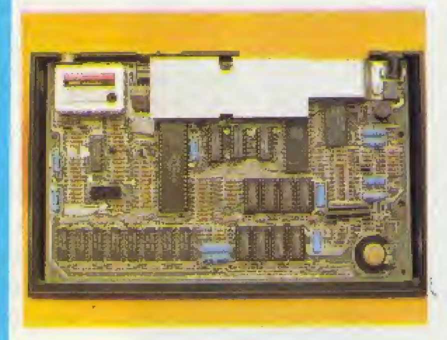
Administración: INFODIS, S. A.
Gerente de Circulación y ventas: Luis Carrero
Producción: Miguel Onieva
Director de Marketing: Antonio González
Servicio al cliente: Julia González. Tel. 733 79 69
Administración: Miguel Atance y Antonio Torres
Jefe de Publicidad María José Martín
Dirección y redacción: Bravo Murillo, 377-5.º A. Tel. 733 74 13
Telex: 48877 OPZX e 28020 Madrid
Administración y Publicidad: Bravo Murillo, 377-3 E. Tels. 733 96 62/96
Publicidad Barcelona: María del Carmen Ríos, Jorge González. Pelayo, 12. Tel. (93) 318 02 89. 08001 Barcelona.
Depósito legal: M-29041-1984
Distribuye S.G.E.L. Avda. Valdelaparra, s/n. Alcobendas-Madrid.
Fotomecánica: Karmat, C/ Pantoja, 10. Madrid.
Fotocomposición: Artecomp.
Imprime: Héroes, C/ Torrelara, 8. Madrid.
Esta publicación es miembro de la Asociación de Revistas de Información asociada a la Federación Internacional de Prensa Periódica, FIPP.

SUSCRIPCIONES:
Rogamos dirijan toda la correspondencia relacionada con suscripciones a:
TODOSPECTRUM
EDISA: Tel. 415 97 12
C/ López de Hoyos, 141-5.º
28002 MADRID
(Para todos los pagos reseñar solamente TODOSPECTRUM)

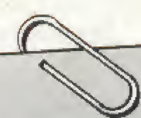
Para la compra de ejemplares atrasados diríjanse a la propia editorial
TODOSPECTRUM
C/ Bravo Murillo, 377-5.º A
Tel. 733 74 13-28020 MADRID

AÑO I • NUMERO 8



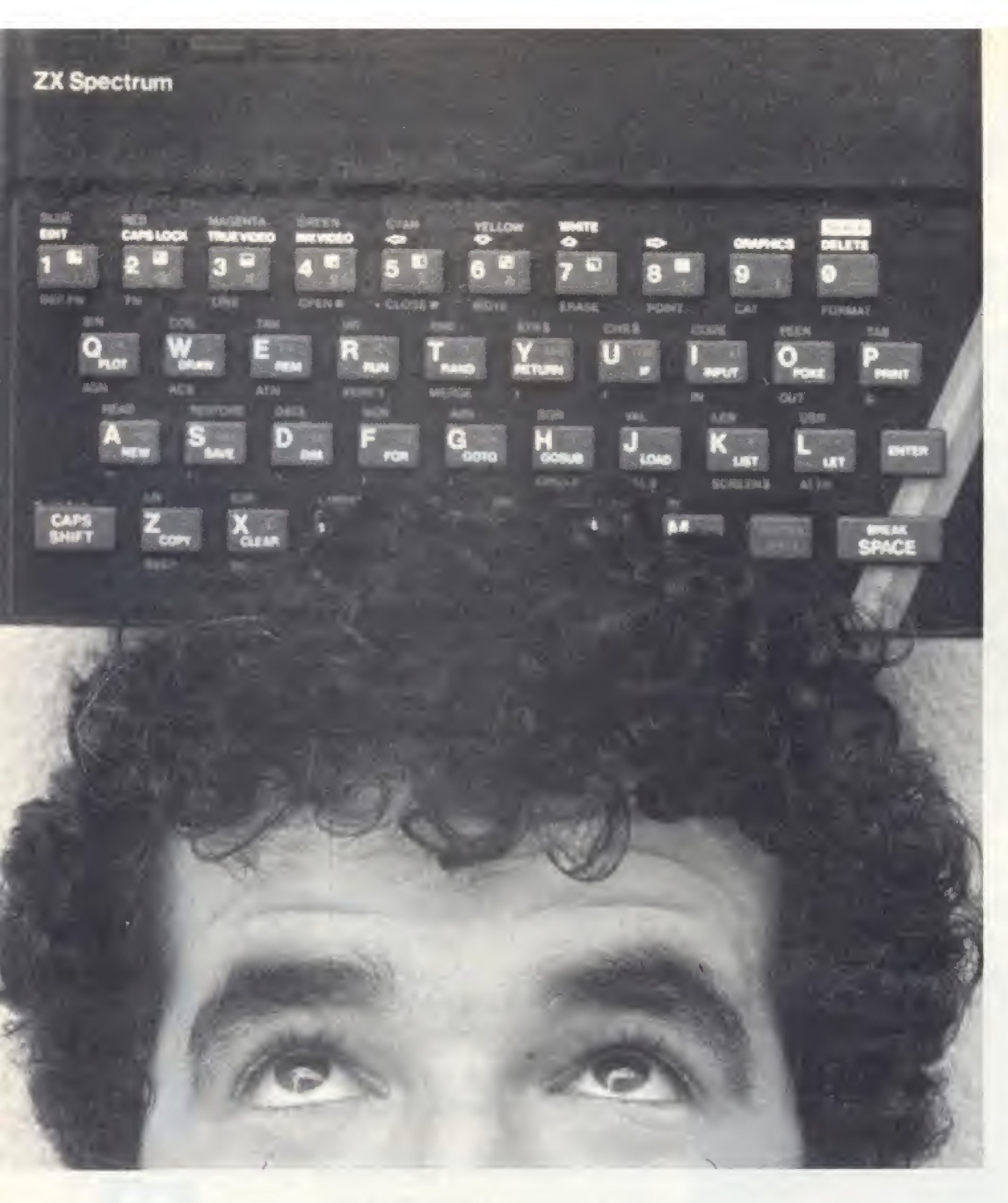






Si tuviésemos que calificar este mes de alguna forma, diríamos que este es el mes del QL. La gran expectación contrasta sin embargo con el desconocimiento que existe sobre este ordenador, lo que dice bastante de la buena campaña de imagen que Sinclair imprime a todos sus productos. A él dedicamos las páginas centrales, sin por ello abondonar a nuestro querido Spectrum.

Hasta el próximo número.



Al bloque de memoria acceden el bus de direcciones *(address bus)*, el *bus* de datos *(data bus)* y las señales de control que son WR (escritura), RD (lectura), y MREQ (requerimiento de memoria)..

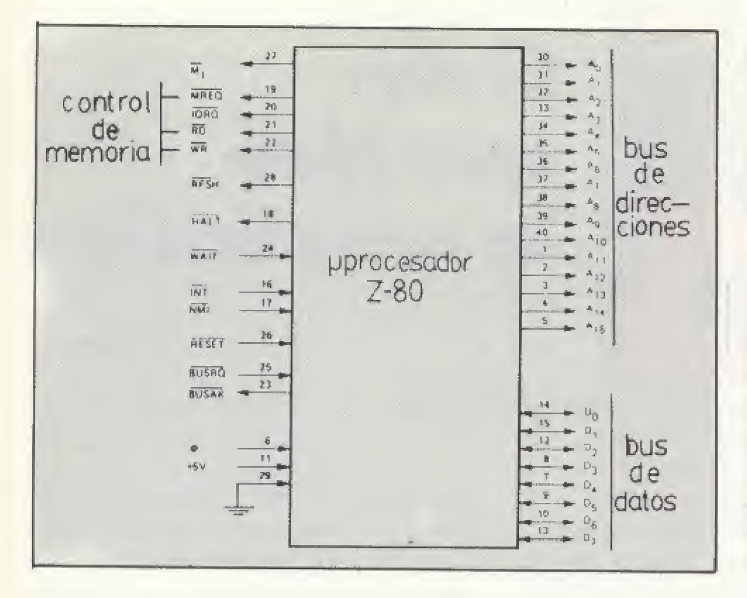
El número de bits del *bus* de direcciones nos dará el espacio máximo direccionable, es decir, el número máximo de bytes a los que podemos acceder. El Spectrum es un ordenador desarrollado a partir de un microprocesador muy popular (y económico): el Z-80; que posee un *bus* de direcciones de 16 bits. Podremos por tanto direccionar $2^{16} = 65.536$ bytes. Complicando tanto al *software* como al *hardware* se puede conseguir direccionar mayor cantidad de memoria a partir del Z-80, formando diferentes «páginas» de 64 Kbytes cada una, pero este no es el caso del Spectrum.

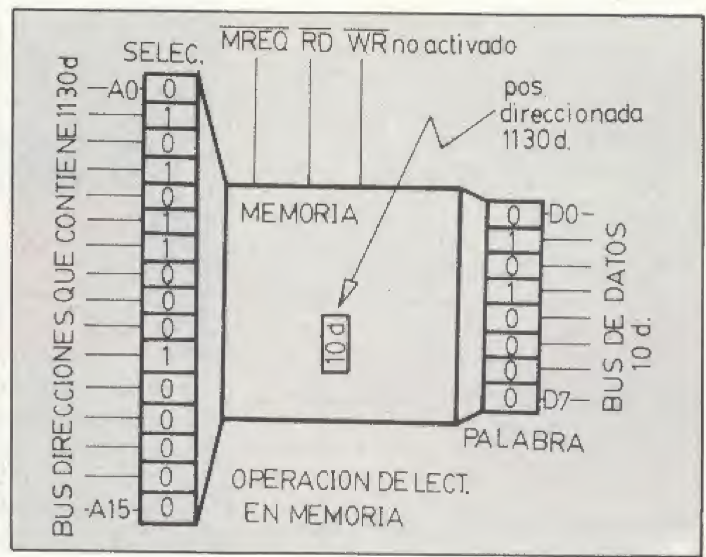
Los 64 Kbytes de espacio direccionable se dividen en dos grupos muy bien diferenciados: los primeros 16 Kbytes están ocupados por memoria de sólo lectura (ROM) y los 48 Kbytes restantes por memoria de lectura y escritura, conocida normalmente con el nombre de RAM, aunque la denominación no es muy correcta. RAM es la abreviatura inglesa de

AMPLIA TU MEMORIA... A

La caída de precios de los últimos meses ha hecho que la mayor parte de los usuarios disfruten del Spectrum de 48 K. Pero no fue este el caso de los «pioneros» Sinclair, quienes a duras penas pudieron acceder al Spectrum 16 K, y que ahora se encuentran discriminados ante el abundante software existente para el 48 K, demasiado «largos» para su reducida memoria.
Afortunadamente este problema ya no es problema si se puede y sabe efectuar la ampliación de memoria correspondiente.

Random Access Memory que significa memoria de acceso aleatorio, y son aquellas a las que se puede acceder a una palabra (byte si ésta es de 8 bits) de una manera directa, siendo el tiempo de acceso independiente de la posición de la palabra a acceder. Por el contrario existen memorias de acceso secuencial, en las que para acceder a una palabra es necesario pasar antes por todas las anteriores. La idea es similar a la diferencia entre el disco y la cinta de *cassette:* en el disco no hace falta rebobinar, se accede directamente. Según lo anterior la ROM también es RAM aunque a lo largo del artículo se utilizará este tér-





mino con su significado usual, es decir, como memoria de lectura y escritura.

Este segundo bloque sólo tiene 16 Kbytes en el Spectrum «pequeño», por lo que habrá que incrementarlo en 32 Kbytes para conseguir el mayor.

En el bloque de sólo lectura se almacenan los programas que permiten al Spectrum empezar a funcionar nada más dar tensión a sus circuitos, tales como las rutinas de grabación, las de impresión de caracteres en pantalla, el intérprete de BASIC, etc. Así por ejemplo la dirección ØØØØ h., Ø decimal es la apuntada por el microprocesador tras producirse un reset; o la dirección Ø385 h, 949 decimal es la de comienzo de la rutina de sonido.

En el segundo bloque se incluyen las variables del sistema, el fichero de pantalla, los mapas de *Microdrive,* los gráficos definidos por el usuario, etc, y además los programas tanto en BASIC como en código máquina.

Una vez direccionada la memoria, se podrá realizar una operación de lectura: un byte será puesto en el *bus* de datos; o una operación de escritura; el contenido del *bus* de datos será almacenado en una palabra de memoria. Operaciones que se encuentran controladas por tres señales, las tres activas a nivel bajo:

MREQ *(memory request)* que valida el contenido del *bus* de direcciones, e indica que se trata de una operación de memoria, no de entrada o salida desde un periférico. También interviene en el «refresco» de los 32 Kbytes últimos de memoria en el caso del Spectrum.

RD *(read)* que indica a la memoria o a cualquier dispositivo de entrada o salida, que el Z-80 quiere leer un dato que deberá ser puesto en el *bus* de datos.

WR *(write)* que indica que el contenido del bus de datos debe ser almacenado por la memoria o por cualquier otro periférico.

Realización práctica

Primero hay que conseguir los circuitos integrados necesarios que, o bien se pueden adquirir sueltos (lo que a veces no es muy fácil en el mercado español) en tiendas de componentes electrónicos, o en forma de kit.

Los multiplexores son 2 intregrados 74 LS 157, las puertas NAND se encuentran en el integrado 74 LS 00 y las OR en el 74 LS 32. En cuanto a las memorias, ya sabemos que se trata de 8 integrados TMS 4532 L3 u ocho integrados TMS 4532 L4, todos de la misma clase. La ampliación no se podrá realizar si se mezclan ambos tipos.

Para instalarlos se extraen los 5 tornillos que unen la carcasa y el teclado del ordenador. Para ello hay que volver el Spectrum boca abajo.

Una vez extraidos se vuelve el ordenador a su posición normal y se retira el teclado suavemente, sin tirones, para no dañar los conectores que son muy delicados.

Después se afloja el tornillo pequeño que se encuentra en el medio de la placa de circuito impreso y se extrae éste. A continuación se afloja otro tornillo que encima del regulador de tensión sujeta al refrigerante a la placa de circuito impreso. Una vez quitado el refrigerante queda totalmente al descubierto la zona de ampliación de memoria.

La operación anterior corresponde al caso de la versión 3B (ISSUE 3B). Si el Spectrum es de la versión 2 (ISSUE 2) no es necesario retirar el circuito impreso ni el refrigerante.

Ahora se realizan los puentes. En la versión 3B hay que unir los puntos

48K

marcados como «TL» por un lado y los marcados como «3» por otro, si la RAM es de código TMS 4532 L3; y los puntos «T1» y «4» si el código es TMS 4532 L4.

Para el ISSUE 2 hay que realizar un solo puente entre el punto marcado por LINK y 0 voltios si la RAM es de la versión TMS 4532 L3; o entre LINK y 5 voltios si el código de los chips es TMS 4532 L4.

A continuación se insertan los integrados, quedando la expansión totalmente realizada.

Memorias estáticas y dinámicas de lectura y escritura

Las memorias estáticas son aquellas que almacenan información sólo con que los circuitos estén alimentados y los accesos de la información sean correctos. Se encuentran formadas por biestables y ocupan un área muy grande en el proceso de integración.

Las memorias dinámicas poseen menos transistores por bit almacenado, y ocupan menos área en el chip, permitiendo memorias más grandes, más baratas y que consumen menos potencia. Pero tienen una desagradable desventaja: olvidan. Es necesario proceder a su refresco cada un cierto tiempo. No están formadas por biestables y mantienen la información gracias a las capacidades (parásitas para otro tipo de aplicaciones) que aparecen en los transistores MOS. En el caso de las memorias, la capacidad cargada almacena un 1, y descargada un 0. Aunque el aislamiento es muy grande, siempre hay pequeñas corrientes de fugas que descargan estas capacidades, y por ello la información se pierde si no hay refrescado.

El tipo de tecnología utilizada: MOS o bipolar condiciona el tiempo de acceso, el precio y la capacidad. Las memorias bipolares son muy rápidas pero caras y de pequeña capacidad. Las MOS son más lentas, baratas y de mayor capacidad. Las memorias de lectura y escritura empleadas en el Spectrum son dinámicas y de tecnología MOS.

La memoria ROM

Bajo este nombre genérico se agru-



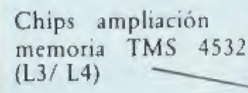
El día que hice la ampliación

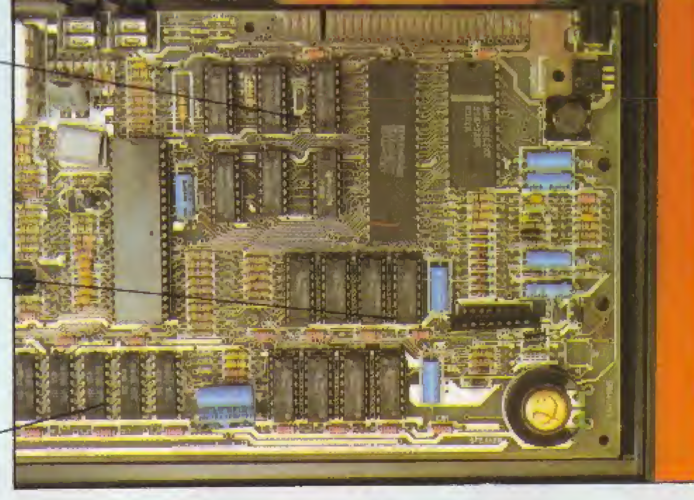
Quienes sabiendo muy poco acerca del «mundillo» de los microordenadores o por razones económicas optamos por comprar un Spectrum de 16 Kbytes de memoria RAM, pronto nos encontramos con la desagradable sorpresa de que la mayoría del *software* comercializado y prácticamente todos los programas de juegos de alta calidad, no podía ser cargado en el ordenador: necesitabamos un Spectrum de 48 *Kbytes*.

Efectivamente, con 16 *Kbytes* de memoria (16 * 1024 *bytes;* 1 *Kbyte* = 1024 *bytes*) el Spectrum sólo puede almacenar programas muy cortos, además no podemos utilizar íntegramente esa capacidad de memoria, parte de ella reservada para funciones internas del propio sistema, como el fichero ded representación visual, los atributos y las variables del sistema entre otras. Nos quedan entonces aproximadamente 8 *Kbytes* de memoria utilizables directamente, lo que para una gran parte de aplicaciones es demasiado poco.

Existen técnicas especiales con las que se consigue mayor capacidad de memoria. Sabía de la existencia de algunos copiadores que conviven simultáneamente con el programa que va a ser duplicado aunque este ocupe el máximo de memoria posible. Este tipo de programas se ubican en el fichero de representación visual haciendose transparente en la pantalla del monitor, pero inutilizándola parcialmente, según me habían dicho.

Es evidente que esta clase de técnicas sólo son utilizadas por el programador experimentado, qué comprende con mayor fuerza que el neófito los inconvenientes que origina una memoria de corta longitud.





Planteado el problema me pregunté cómo incrementar esa memoria en los 32 *Kbytes* necesarios para conseguir las prestaciones de su hermano mayor: el Spectrum de 48 *Kbytes*. Así que me informé.

En principio uno y otro modelo tienen como única diferencia una serie de circuitos integrados que en la versión de 48 *Kbytes* se encuentran soldados en la placa de circuito impreso, y en la versión de 16 *Kbytes* se encuentran los zócalos pero no los circuitos integrados.

Armado con esta información, pude conseguir los circuitos integrados necesarios, y los conecté tomando

como modelo un Spectrum de 48 *Kbytes* de la misma versión (ISSUE 3-B) que el mío de 16 *Kbytes*. Después de colocar los integrados alimenté el ordenador y sorprendentemente no funcionaba. En la pantalla del televisor correctamente sintonizado no aparecía el mensaje de *copyright* de Sinclair; muy al contrario, el Spectrum no enviaba ningún tipo de señal que el receptor de televisión pudiera visualizar.

Con gran temor de haber averiado la circuitería del Spectrum desconecté los integrados que antes había insertado y alimenté de nuevo el ordenador. Todo funcionaba normalmente.

Después de esta nefasta experiencia decidí llevar el Spectrum a una tienda especializada del ramo, donde después de intentarlo, me confirmaron que el ordenador no admitía la ampliación interna, y me aconsejaron que, o bien adquiría una ampliación externa, o lo ponía en manos del servicio de reparación de **Investrónica.**

Decidido a tener un Spectrum sin ampliación externa ya que había visto con mis propios ojos el «hueco» que tenía el circuito impreso para la ampliación interna, me decidí por enviarlo al servicio de reparaciones de **Investrónica.**

Por fin y después de abonar 9.975 pts. conseguí tener mi Spectrum ampliado a 48 *Kbytes* de memoria RAM.

Más adelante descubrí que la posible causa de que no funcionase el ordenador con la ampliación realizada por mi mismo, podría estar en una serie de puentes que hay que realizar en la versión pequeña para convertirla en la de 48 *Kbytes*.

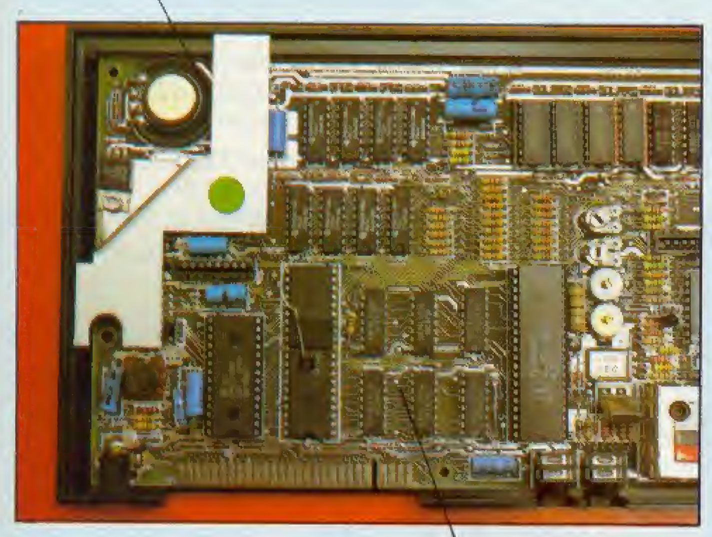
Sin embargo a aquellos que no tengáis una mínima experiencia en el ensamblado de circuitos electrónicos, os aconsejo no realizar la ampliación con vuestras propias manos y, aunque sea un poco más caro confiar la operación a cualquier tienda especializada, o al propio servicio de reparaciones de **Investrónica.**

Javier Sánchez

Vista parcial issue 2, destacando la ampliación.

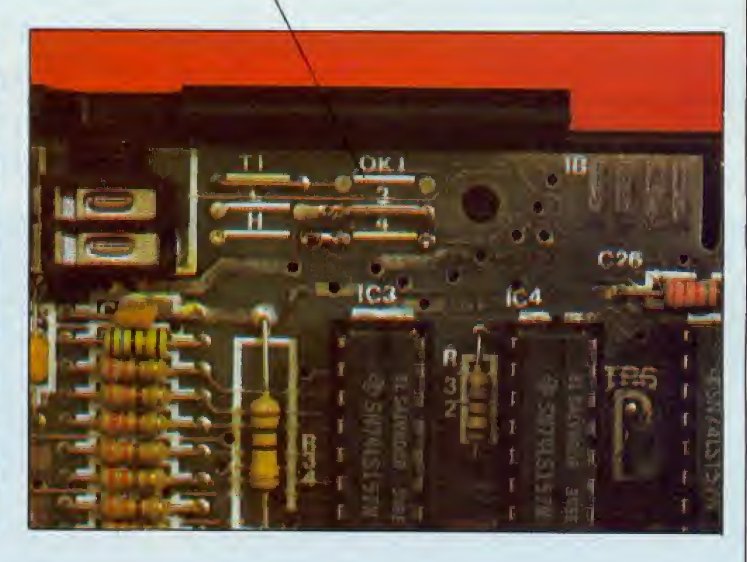
Refrigerante del regulador de tensión.

TMS 4532 (L3/L4)

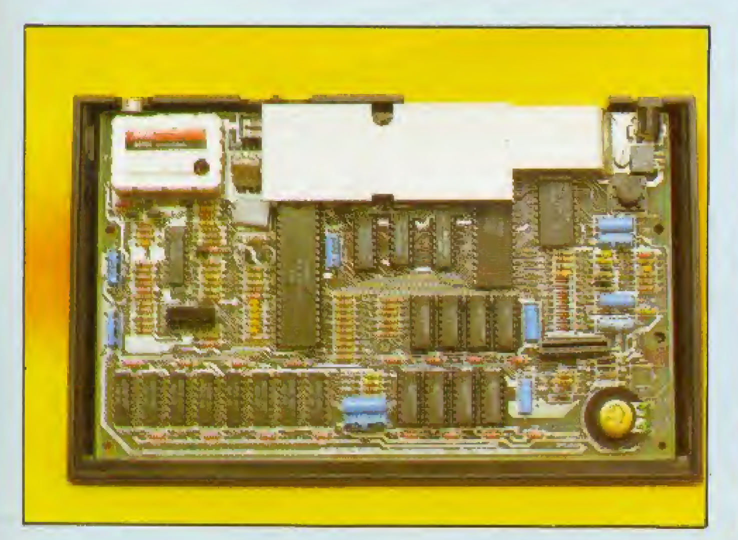


control de la memoria

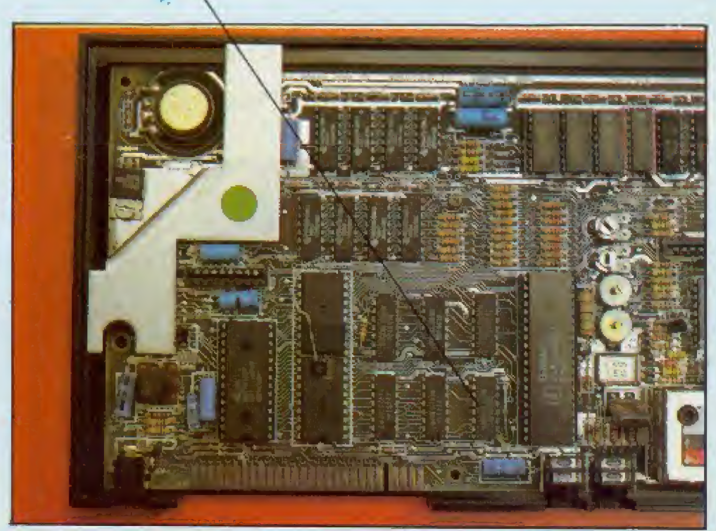
Puentes issue 3B para el modelo de memoria TMS 4532 (L3).



Issue 3B sin retirar el refrigerante del regulador de tensión.



Zona de puentes issue 2.



pan tres clases de memorias cuyo contenido es inalterable para el programador:

ROM propiamente dicha.- Su contenido es completamente inalterable desde la fabricación del chip.

PROM.- Su contenido se programa una vez y después permanece inalterable. Las modificaciones realizadas en la grabación son destructivas.

RPROM.- Son de tecnología MOS y su contenido se puede programar y permanecer inalterable para el programador que maneja el ordenador. Pero pueden volver a ser puestas en blanco y reprogramadas. El proceso de grabación no es destructivo.

Cualquier usuario con suficientes conocimientos puede cambiar la ROM del Spectrum por una PROM o una RPROM e intentar corregir, en el nuevo chip, los defectos de programación, de algunas rutinas del diseño original de Sinclair.

Como funciona la expansión

Las memorias de 32 Kbytes utilizadas en la ampliación son la TMS 4532 L3 o la TMS 4532 L4. Cada chip tiene 16 pines y almacena el contenido de un solo bit en cada posición direccionable. Por lo tanto se necesitan 8 circuitos integrados para almacenar 1 byte.

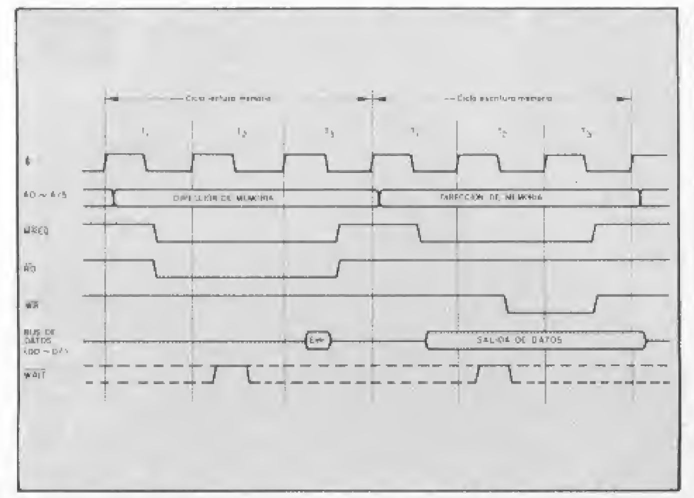
Si el *bus* de direcciones es de 16 bits y hay 16 terminales en cada chip da la impresión de que no hay suficientes pines para las señales de control de memoria, así como para la alimentación. En realidad sólo 8 de esos 16 terminales están dedicados a la selección de la dirección. Los 16 bits se introducen multiplexados en 2 grupos de 8 bits, de tal manera que primero se introducen los 7 bits correspondientes al direccionamiento de una fila de la memoria, y después los correspondientes a una columna. Dos grupos de 8 bits suponen 16 en total y sin embargo sólo son necesarios 15(8 + 7), por lo tanto sobra uno. Este bit, A(15), se pone a 0 o a 1 en función del integrado que hayamos adquirido, realizando, mediante trozos de hilo conductor, unos puentes en el circuito impreso del Spectrum.

Si tenemos el TMS 4532 L3 se debe poner a 0, y en el caso del TMS 4532 L4 debe de estar a 1. Para entender el funcionamiento basta seguir el esquema del circuito y el cronograma de una operación de memoria.

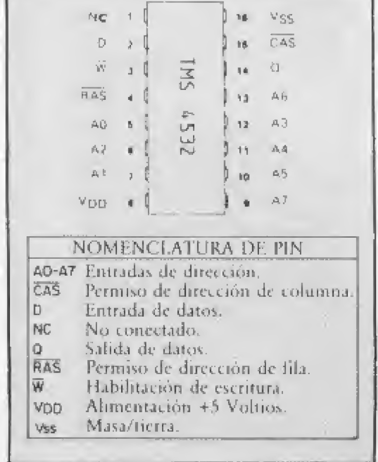
Si no domina el ensamblado de circuitos electrónicos, mejor déjeselo a un especialista.

Primeramente el microprocesador coloca en el *bus* de direcciones la posición del byte a seleccionar. Si se trata de una dirección perteneciente a los 32 Kbytes últimos de memoria, es decir, una dirección de la ampliación, A(15) se pone a nivel alto. Simultáneamente MREQ, RD, y WR están también a 1. A la salida de P5 aparece también un 1 con lo que los pines S (selección) de los circuitos integrados IC 26 y IC 25 que son multiplexores (computadores electrónicos digitales), están también a nivel alto. Estos circuitos presentan a la salida (referencia Y) lo que se encuentra en las entradas marcadas por A si S está a 0, o lo que haya en las entradas de referencia B si S está a nivel alto. Aparecen entonces a la salida de los multiplexores los bits A0, A1, A2, A3, A4, A5 y A6 correspondientes a la selección de una dirección de fila en la memoria.

Con medio ciclo de reloj de retardo se activan MREQ y, si se trata de una operación de lectura también RD (pasan a nivel bajo). WR continúa a 1 y por lo tanto en la entrada WR de las memorias existirá ese mismo nivel indicando una operación de lectura. La señal a la salida de P4 cambia a nivel bajo unos 20 ns. después, y aproximadamente 10 ns. antes ya se había activado la entrada RAS a través de P2. La activación de RAS *(row address strobe* = permiso de dirección de fila), valida lo que hay en las entradas A0, A1, A2, A3, A4, A5, A6 y A7 de los chips de la RAM como la dirección de una fila de memoria. El pin S de los multiplexores tarda más en cambiar de estado debido al retardo de aproximadamente 35 nanosegundos, introducido por A71-C63, P2 y P5, a partir de la activación de MREQ y RD.

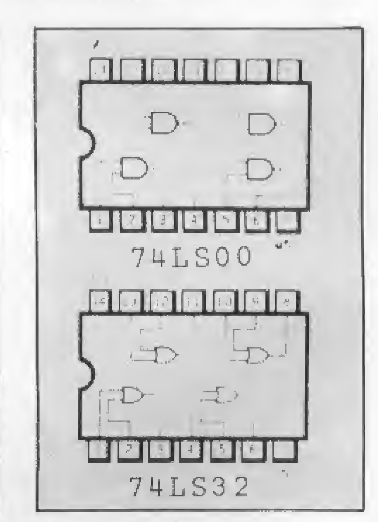


Cronograma de una operación de memoria.



NOMENCLATURA DE PIN	
AO-A7	Entradas de dirección.
CAS	Permiso de dirección de columna.
D	Entrada de datos.
NC	No conectado.
Q	Salida de datos.
RAS	Permiso de dirección de fila.
W	Habilitación de escritura.
VDD	Alimentación +5 Voltios.
Vss	Masa/tierra.

Descripción de patillaje de los integrados de la RAM.



Descripción de patillaje. 74 LS 00 74 LS 32



El tiempo que transcurre entre la activación de RAS Y EL PASO A 0 de 8 es suficiente para la entrada de la dirección de la fila.

Con otro retardo de unos 35 ns. y a través de P6, se produce la activación de CAS (*column address strobe* = permiso de dirección de columna) que valida el contenido de las entradas como la dirección de una columna. Recordemos que el terminal S de los multiplexores habían pasado a 0, con lo que en las salidas de estos integrados se encontraban los bits A7, A8, A9, A10, A11, A12, A13 y A14 de la dirección.

Realmente las entradas de las direcciones de las filas y de las columnas en la memoria se producen con el flanco de bajada de las correspondientes señales de control, por lo que

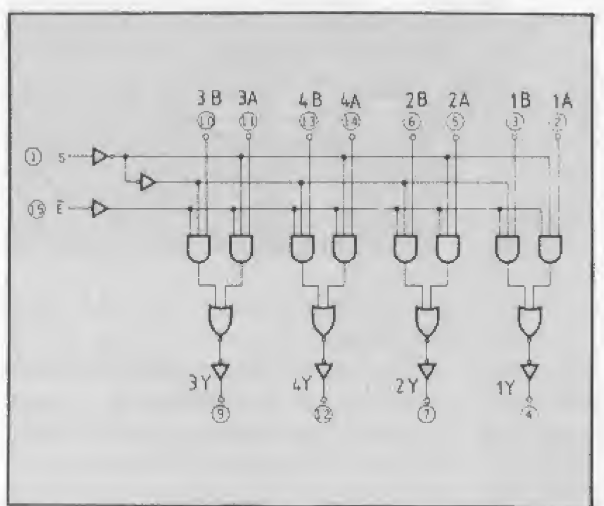
tiempos muy pequeños son suficientes para que se produzca la transferencia de bits.

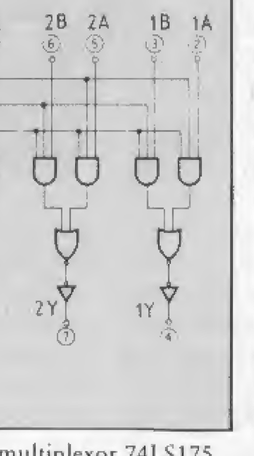
La operación de escritura es similar, y el lector con conocimientos medianos de electrónica digital la podrá seguir a través del esquema ayudándose con el cronograma antes mencionado.

Activando la señal RAS se consigue el refrescado de las 128 direcciones de fila, es decir, proporciona la regeneración de todos los bits en cada una de las filas. El refrescado se debe hacer al menos una vez cada 2 ms., y mientras se está realizando se anula cualquier salida. El Spectrum produce esta operación cada vez que se activa MREQ, a través de P2, que actúa como *buffer* de la entrada RAS.

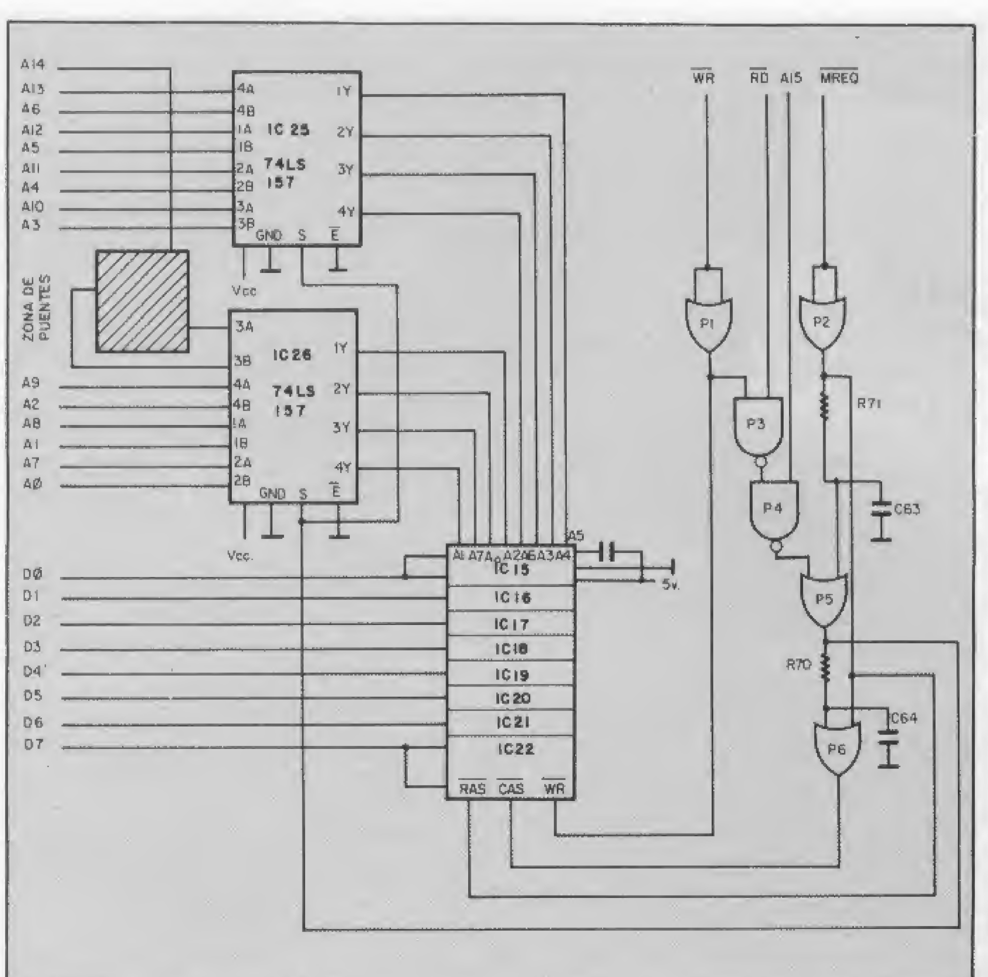
Para comprobar que la ampliación ha sido bien instalada basta con teclear PRINT PEEK 23733 y pulsar ENTER. Si la operación ha sido correcta aparecerá en la pantalla el número 255. Si aparece un 127 el ordenador está trabajando con 16 Kbytes de RAM y la expansión no ha sido realizada con éxito. Realmente lo que hacemos es comprobar el contenido del byte de mayor peso de la variable P-RAM, en donde se almacena la dirección del último octeto de la memoria física.

Por último queda por comentar que se puede realizar la ampliación con RAMs de 64 Kbytes utilizando sólo la mitad de su posibilidad de almacenamiento. Para ello hay que replanificar algunos puentes o conexiones, y sólo deben intentarlo aquellos que sean bastante expertos en electrónica. RAMs de 64 Kbytes válidos son la TMS 4564 y la TMS 4164 que probablemente sean más fácil de conseguir que los mencionados anteriormente.

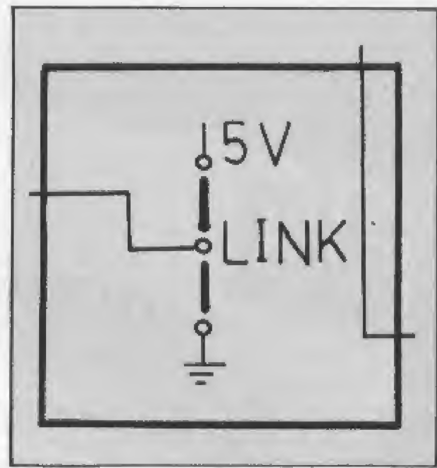




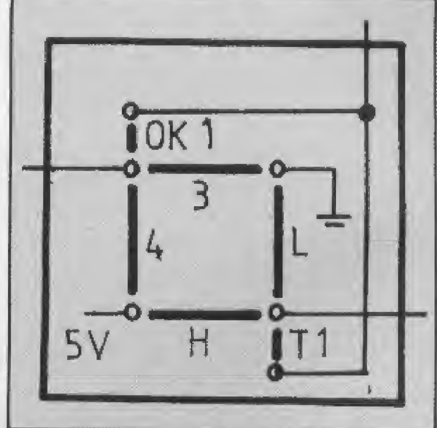
Descripción interna y patillaje del multiplexor 74LS175.



Esquema de la expansión de memoria



Zona de puentes ampliada issue 2.



Zona de puentes ampliada issue 3B.

SPECTRUM COMPUTING

DOS GRANDES JUEGOS EN CODIGO CON MAQUINA OPCION DE JOYSTICK

JUEGOS

Chopper PILOTANDO UN HELICOPTERO, TENDRA QUE DIRIGIR EL EQUIPO DE RESCATE PARA SALVAR A LOS ABANDONADOS EN UN CAMPO PETROLIFERO, DE UNA MUERTE SEGURA.

Convoy DESTRUIR LOS ALIENIGENAS Y SUS NAVES ES LA UNICA FORMA DE SALVAR LA TIERRA DE LA INVASION.

ARTICULOS

Twiddler MUESTRA LOS MISTERIOS DEL RAPIDO CAMBIO DE COLOR.

Cartoon APRENDA A PROGRAMAR DIBUJOS ANIMADOS.

Hangout CONOZCA MAS A FONDO LAS POSIBILIDADES DE SU ORDENADOR.



y mucho más

SORTEO ESPECIAL

MAS DE 150.000 PTAS. EN PREMIOS BASES EN EL INTERIOR



PREYME es, sin duda, una de las aplicaciones en *microdrive* más serias que hemos podido encontrar para el Spectrum de 48 *Kbytes* (¡seguro que no la última!), especialmente para el campo de la arquitectura, campo generalmente ocupado por ordenadores mayores y periféricos especiales como los *plotters* o "trazadores". En suma, aplicaciones generalmente caras. Este programa permite realizar este mismo trabajo, con las lógicas limitaciones de las 48 *Kbytes* de memoria.

Destaca en primer lugar su cuidada documentación. Un buen manual con el que "enfrentarse" a los cuatro *microdrives* que integran el paquete. Los programas están grabados en el *microdrive* PREYME, al que se le suman los de UNITARIOS, DESCOMPUESTOS Y OBRAS. Es decir, un cartucho para los programas y tres para archivos de datos, a utilizar según las opciones que se requieran, teniendo en cuenta que en ocasiones se necesitará trabajar simultáneamente con dos cartuchos, por lo que necesitará contar —además del interface 1— con dos unidades de *microdrive.*

Para aprovechar mejor la memoria, PREYME es en realidad un paquete integrado de cuatro programas distintos (Unitarios, Descompuestos, Obra y Varios), desde los cuales se accede a los datos grabados en los archivos antes mencionados.

El programa Unitarios maneja fundamentalmente el archivo de unitarios, pudiendo acceder al de Obra cuando lo necesita. El programa Descompuestos utiliza el archivo de unitarios para la formación de descompuestos o el de obra, para ver los descompuestos que intervienen en una obra determinda. El programa Obra realiza el presupuesto de la obra. Y, finalmente, el programa Varios realiza diversas funciones como la inicialización del cartucho, visualización de la memoria disponible y copia de seguridad.

La figura 1 muestra estas cuatro opciones, tal y como salen en pantalla tras la operación de carga del programa. El programa se sitúa automáticamente en el bloque tercero (Obra), como puede verse por el fondo de color oscuro que identifica el bloque de programa cargado en memoria. Después se pide el código referente a la opción a utilizar.

Antes de elegir opción fíjese en la información que ofrecen las dos últimas columnas. Ya que no se puede contener toda esta información en memoria y se trabaja con dos *microdrives* simultáneamente, las dos últimas columnas informan de los cartuchos que habrán de estar presentes en el 1 y en el 2. Es inevitable el intercambio continuado, pero al menos se informa adecuadamente, aunque al principio siempre cometerá algún error. (Si este es su caso, ejecute GOTO 1, con lo que le aparecerá un mensaje de error, pasando después al menú.) Veamos resumida-

APLICACIONES EN ARQUITECTURA

PREYME: Presupuestos y mediciones

Si usted es arquitecto o aficionado a la arquitectura, más de una vez habrá tenido que hacer un presupuesto, es decir, pasarse toda la noche calculando y escribiendo en su vieja máquina de escribir para poderlo presentar a tiempo... ¡Y sin errores! Si a su afición por la arquitectura se une un interés por la informática y posee un Spectrum, sus noches en vela habrán terminado.

mente los cuatro bloques en que se divide el programa.

Unitario

El fichero puede contener hasta 32 capítulos con 32 unitarios cada uno, 1.024 en total. Cada unitario está definido por el número de capítulo y el orden dentro de dicho capítulo, cuatro cifras que lo referencian para llamadas posteriores desde los descompuestos. El número máximo de caracteres es de 100 para la descripción de la unidad y 9 para el precio (ver figura 2).

Las opciones de este programa son: consultas y modificaciones, listado unitarios, materiales de obra y revisión de precios.

La consulta y modificación tiene una cómoda y bonita presentación, utilizándose las teclas de la 1 a la 9 para avanzar el número de unidades equivalente al número de la tecla pulsada.

La revisión de precios unitarios es automática, de acuerdo a las dos listas de precios existentes en archivo que se suministran con el programa, una de las cuales permanece con sus valores a cero. Las diversas operaciones relacionadas con esta opción son sumamente lentas, debido al acceso a microdrive.

Maneja tres archivos de datos: Obras, Descompuestos y Unitarios

Descompuestos

Al igual que el de Unitarios, puede contener un máximo de 32 capítulos con 32 elementos cada uno, 1.024 en total.

Las opciones de este programa son: consultas y modificaciones, listado de descompuestos, cuadro oficial de precios y actualización de precios.

Las consultas y modificaciones se realizan por el mismo procedimiento que en Unitarios.

Las opciones de listado de descompuestos y cuadro oficial se dan por impresora. Proceso ligeramente lento, especialmente en el primero, ya que la búsqueda de datos ha de hacerse en dos archivos simultáneamente. El cuadro oficial de precios es similar, imprimiendo sólo las partidas que entran a formar parte de una obra. Finalmente, la actualización de precios se realiza sobre la lista de precios unitarios.

Obra

Cada nueva obra necesita un cartucho con el nombre de "Obra"

PREYME

e inicializado con la opción "13 inicializar" que veremos en el siguiente programa. El archivo de Obra es similar al de Descompuestos, contando con 32 capítulos de 32 partidas cada uno (ver figura 3).

Las opciones de este programa son: consultas y modificaciones, presupuesto, decodificar y actualización de precios.

En realidad es un paquete integrado, compuesto de cuatro programas distintos

La información de cada partida de obra es de un código (cuatro cifras), descripción (introducción automática), mediaciones y comentarios (12 líneas), e importe (calculado automáticamente con los datos de la medición y precio elegido).



Figura 1. Menú de opciones. La zona en negro indica los programas residentes en memoria.

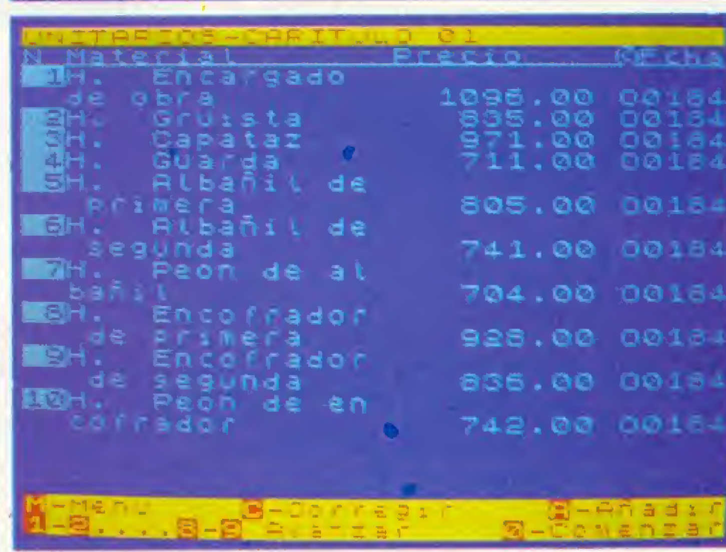


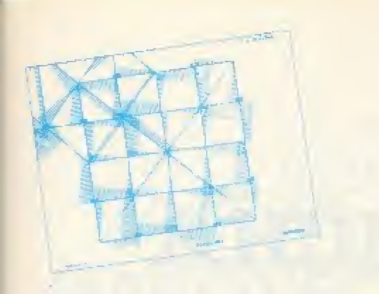
Figura 2. Ejemplo de un capítulo de unitarios.

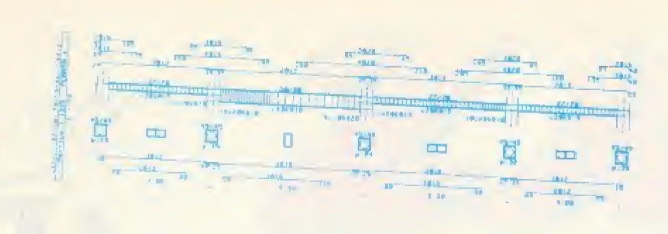


LO QUE DEBE TENER UN PROGRA

Para hablar de programas de mediciones, nadie mejor que Fernando de Valderrama, arquitecto y autor del programa PRESTO, uno de los primeros programas profesionales aparecidos en el mercado para grandes ordenadores. Rodeado de equipos **Hewlett-Packard** 150, nos cuenta "lo que debe tener un programa de mediciones", que resume en tres características: la dimensión adecuada de los archivos, la flexibilidad de uso y la rapidez.

"Un fichero de precios debe tener 600 partidas como mínimo. Los bancos de datos estándar tienen 1.200 partidas, como es el caso de la Junta de Andalucía, o de 3.000 para la Generalitat. Pero 600 es una buena cifra para poder manejar la información de forma eficiente. Esto en cuanto a obra, ya que también hay que tener en cuenta que en cada obra se manejan 4.000 líneas de medición, es decir, las líneas en las que se descompone la medición de la unidad de obra. Manejar 4.000 líneas es lo normal. Y medido en capítulos, el





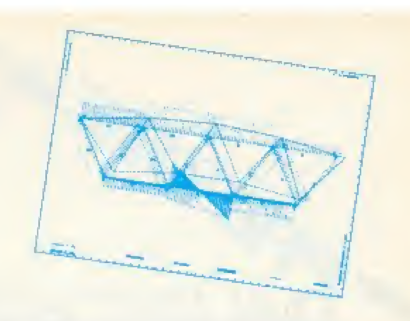




Figura 3. Ejemplo de un capítulo de obra.

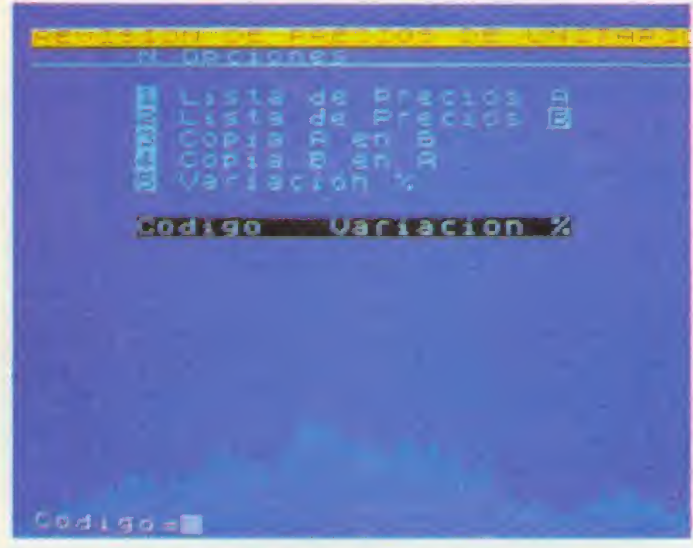


Figura 4. Revisión de precios, una de las opciones más lentas.

Varios

El cuarto programa consiste en cuatro utilidades: inicializar, memoria libre del cartucho, copia de seguridad y definición de impresora.

El proceso de inicialización de cartuchos consiste realmente en dos pasos: formateado de un cartucho e inicialización para trabajar con PREYME.

La opción Memoria libre informa de la capacidad disponible en *Kbytes* del cartucho elegido. Tampoco es demasiado esfuerzo si se tiene en cuenta que este dato ya se proporciona al pedir el catálogo de cualquier cartucho. Sería más útil que la información estuviese referida al número de unidades contenidas.

La definición de impresora posibilita la obtención de resultados por diversos modelos del mercado y en distintos formatos de impresión.

Resumiendo

Este programa se enmarca dentro de las denominadas aplicaciones verticales, y de su seriedad da

MA DE MEDICIONES

número normalmente empleado es de 30, con 40 ó 50 partidas por capítulo.

Pero lo de menos son las cifras. Con el PRESTO se puede hacer desde un chalet hasta una autopista. A veces se hacen programas a medida porque *la gente no se da cuenta de hasta qué punto trabaja con limitaciones.* También hay quienes creen que el capítulo 1 siempre tiene que ser el del movimiento de tierras. Todos los datos han de ser totalmente modificables. Por ejemplo, la asignación de la grúa a la obra, normalmente se aplica el 3 %, pero el Ministerio de Obras Públicas prefiere un 1 y la Junta de Andalucía el 8,1 de gastos auxiliares. En resumen, el usuario debe poder estructurar su programa de acuerdo a sus necesidades.

Finalmente, el programa ha de ser rápido. Si obliga a mantener un esquema demasiado rígido no sirve. Generalmente este tipo de trabajos se hacen al final en una noche. Con un buen programa se puede empezar más tarde para afinar más y acabar al mismo tiempo o antes que la competencia."

Lógicamente el PRESTO no es compatible con el Spectrum, preparado para equipos de 120 K de memoria y dos floppys de 270 Kbytes cada uno. El precio varía de 200.000 a 260.000, según opciones. Prestaciones y precios por encima de lo que estamos acostumbrados en el Spectrum. Una buena comparación para conocer mejor al Spectrum, y por ende, aprovechar mejor sus capacidades.



buena prueba el que sea distribuido por el Colegio de Arquitectos de Madrid. Hasta ahora, la informatización de este trabajo ha requerido fuertes inversiones en ordenadores más grandes y en programas más complejos. En contrapartida, las dimensiones de sus ficheros son más limitadas y su manejo más incómodo: en vez de trabajar con un disco, hay que utilizar cuatro *microdrives*. Pero si sus actividades no son demasiado grandes, este programa puede resolverle muchos problemas... a un precio interesante.

FICHA

Programa: PREYME.
Ordenador: Spectrum 48K.
Distribuidor: Microgesa.
Periféricos: Interface 1, 2 unidades microdrive e impresora.
Precio: 24.000 ptas.

Figura 5. Ejemplo listado de capítulos unitarios.

CAPITULO 5
Estructuras

No.	Cod.	Ud.	Descripcion	Precio
1	0501	Kg.	Acero corrugado suministrado, elaborado y colocado en pilares	

Cod.	Cantidad	Ud.	Unitario	Precio unitario
0111	0.036	H.	Ferrallista de primera	914.00
0133	0.036	H.	Peon de Ferrallista	742.00
0902	1.00	Kg.	Hierro redondo corrugado 42	39.50
0332	0.0050	Kg.	Alambre recocido n.8 (1.3mm.)	87.00
0134	%		7% de gastos directos	

IMPORTE TOTAL DE LA PARTIDA.................. 106.53

No.	Cod.	Ud.	Descripcion	Precio
2	0502	Kg.	Acero corrugado suministrado, elaborado y colocado en jacenas	

Cod.	Cantidad	Ud.	Unitario	Precio unitario
0111	0.041	H.	Ferrallista de primera	914.00
0133	0.041	H.	Peon de Ferrallista	742.00
0902	1.00	Kg.	Hierro redondo corrugado 42	39.50
0332	0.0050	Kg.	Alambre recocido n.8 (1.3mm.)	89.00
0134	%		7% de gastos directos	

IMPORTE TOTAL DE LA PARTIDA.................. 115.38

No.	Cod.	Ud.	Descripcion	Precio
3	0503	Kg.	Acero corrugado suministrado, elaborado y colocado en losas planas	

Cod.	Cantidad	Ud.	Unitario	Precio unitario
0111	0.037	H.	Ferrallista de primera	914.00
0133	0.037	H.	Peon de Ferrallista	742.00
0902	1.00	Kg.	Hierro redondo corrugado 42	39.50
0332	0.0050	Kg.	Alambre recocido n.8 (1.3mm.)	89.00
0134	%		7% de gastos directos	

IMPORTE TOTAL DE LA PARTIDA.................. 109.30

No.	Cod.	Ud.	Descripcion	Precio
4	0504	Kg.	Acero corrugado suministrado, elaborado y colocado en losas inclinadas	

Cod.	Cantidad	Ud.	Unitario	Precio unitario
0111	0.045	H.	Ferrallista de primera	914.00
0133	0.045	H.	Peon de Ferrallista	742.00
0902	1.00	Kg.	Hierro redondo corrugado 42	39.50
0332	0.0050	Kg.	Alambre recocido n.8 (1.3mm.)	89.00
0134	%		7% de gastos directos	

IMPORTE TOTAL DE LA PARTIDA.................. 122.47

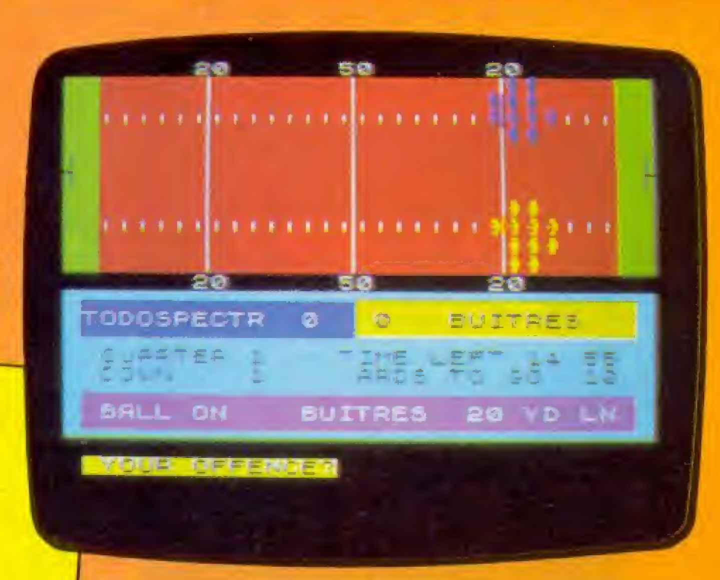
Figura 6. Ejemplo listado de descompuestos.

CAPITULO 5
Aglomerantes y asfaltos

No.	Cod.	Ud.	Descripcion	Precio	Idt	Fecha
1	0501	Kg.	Cemento potland (PA-350) a granel y en obra	6.84	H	01-84
2	0502	Kg.	Cemento portland (PA-350) en saco y en obra.	7.67	H	01-84
3	0503	kg.	Cemento rapido	7.17	H	01-84
4	0504	Kg.	Yeso blanco	3.35	H	01-84
5	0505	Kg.	Yeso comun	2.60	H	01-84
6	0506	Kg.	Cal apagada	11.00	H	01-84
7	0507	Kg.	Cal viva	8.50	H	01-84
8	0508	Kg.	Cemento griffi	10.80	H	01-84
9	0509	Kg.	Cemento adhesivo	35.00	H	01-84
10	0510	Kg.	Cemento Stickfit	38.50	H	01-84
11	0511	Kg.	Cal hidraulica	8.39	H	01-84
12	0512	Kg.	Escayola	4.80	H	01-84
13	0513	M3.	Perlita mezcla con cemento	4779.00	H	01-84
14	0514	Ud.	Saco 501. perlita mezcla con yeso (Perlesca)	314.00	H	01-84
15	0515	Tm.	Betun asfaltico (tipo B 150/220)	29929.00	H	01-84
16	0516	Tm.	Betun asfaltico (tipo b 40/50)	29929.00	H	01-84
17	0517	Tmm	Betun fluidificado (tipo MC-0)	35649.00	H	01-84
18	0518	Tm.	Betun fluidificado (tipo RC-4)	35649.00	H	01-84
19	0519	Tm.	Betun fluidificado en emulsion anionica (tipo EAL) de rotura lenta	24677.00	H	01-84
20	0520	Tm.	Emulsion anionica (tipo EAM-1) de rotura media	28690.00	H	01-84
21	0521	Tm.	Emulsion anionica (tipo EAR-1) de rotura rapida	23035.00	H	01-84
22	0522	Tm.	Emulsion cationica (tipo ECM) de rotura media	28958.00	H	01-84

Juegos

FUTBOL AMERICANO



Los buitres contra TODOSPECTRUM. Los jugadores se agrupan en una esquina.

Distribuidor: Compulogical
Memoria requerida: 48 K
Precio: 1.550 ptas.

GAME STATISTICS
TODOSPECTR 0 0 BUITRES
(PRESS 'P' FOR PRINTOUT)
PRESS ANY KEY TO CONTINUE

Estadística del juego, para tenerlo todo controlado.

No se equivoque con el rugby, en el futbol americano sólo hay 11 jugadores y la pelota es más pequeña. Aunque no exista gran afición por este juego, seguro que las cuotas de adicción aumentan, al menos el juego con Spectrum. No se preocupe si no lo conoce bien, se dan instrucciones completas en las tres pantallas (pantalla de juego, estadísticas y ayuda).

Para comenzar, como en el juego real, se lanza la moneda y el equipo acertante recibe el balón y se convierte en atacante para el que hay unas jugadas determinadas que el defensor tendrá que contrarrestar con jugadas defensivas. Cada jugada atacante tiene su contrapuesta defensiva. Si no se da con la jugada defensiva o atacante correcta se conseguirán pocos o ningún resultado.

El juego es una fiel representación del futbol americano y para acentuarlo más, se han permitido el lujo de visualizar, en cualquier momento del juego, una tabla estadística con todos los porcentajes acerca del desarrollo del juego, práctica común de los partidos americanos.

La representación gráfica es buena, aunque se echa en falta la posibilidad de visualizar el campo de toda la pantalla. Tan sólo ocupa la mitad, dejando la parte inferior de la pantalla para información del juego. De esta forma, los jugadores son demasiado «pequeños» y no hay gran diferencia de color entre los equipos.

Control: Teclado.
Jugadores: Uno.
Gráficos: Buenos colores y definición de gráficos, aunque demasiado reducidos.
Nivel de dificultad: No existen.
Originalidad: Trata de ser un fiel reflejo de futbol americano, incluyendo las largas estadísticas sobre distintos aspectos del juego.
Conclusión: Muy adictivo y complejo, recomendable para fanáticos de este juego.

En cuanto al nivel de dificultad, una vez que haya conseguido hacerse con las reglas del juego, varía con la experiencia de cada uno (aunque la suerte también influye en alguna medida). Cuanto más conozca los distintos tipos de jugadas y el momento de llevarlas a cabo, mayor será su oportunidad de ganar.

Tiene dos opciones de juego: contra el ordenador o contra un tercero. En el caso de que elija al Spectrum como contrario le será bastante difícil ganar. Pero se lo aconsejamos efusivamente, ya se sabe que «para hacer un buen partido hay que estudiar al contrario».

MANAGER

Distribuidor: Software Centre
Memoria requerida: 48 K
Precio: 2.500 ptas.

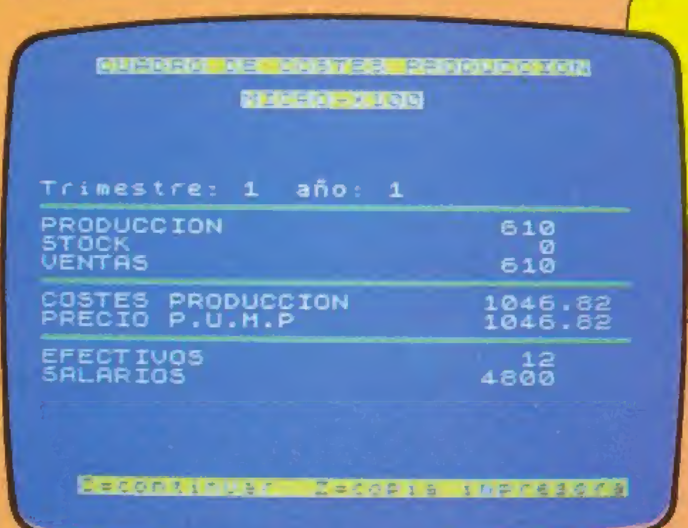
En la facultad de ciencias empresariales de cualquier universidad que se precie, seguro que encontrará un programa similar a este. El ordenador será más grande, pero difícilmente el programa puede ser más completo.

Manager es un simulador de empresa, es decir, su Spectrum le da información de «su empresa» referente a cuatro productos que integran el «proceso productivo»: el microordenador MICRO-XLOO, la unidad de discos FLOP-DISKO, la impresora AZ-PRINTER, el programa de contabilidad LOGICOMPTA, y el de gestión de existencias LOGISTOCKS. Además de otros dos productos en fase de investigación.

Un completo manual de 17 páginas informa de las distintas posibilidades, si bien deja para la última página la forma de comenzar. Al cargar el programa hay que decidir entre JUEGO CARGADO o INICIAR UN NUEVO JUEGO, lo que significa seguir con el resultado grabado de una sesión anterior, o comenzar. Una vez resuelto este pequeño incidente, se puede comenzar a dirigir la empresa, disponiendo de diversas opciones: Visualización de cuadros de resultados, decisiones de producción, decisiones de marketing, decisiones financieras, paso al siguiente ejercicio.

Además, dado que Manager es una traducción de la versión francesa, el juego está adaptado a las normas del Mercado Común, con impuesto I.V.A. Esto puede suponer un poco de confusión al principio, especialmente en la estructura de la cuenta de resultados y del Balance.

La simulación de la realidad empresarial está bien, aunque con «demasiadas» incidencias laborales. Y tenga cuidado con los aumentos salariales: antes de pasar al siguiente ejercicio tendrá que revisar el precio de sus productos, si no quiere vender por debajo de su coste.



Estudio de los costes de un producto. Necesario para fijar los precios.

Análisis del lanzamiento de un nuevo producto.

Control: Teclado.
Jugadores: Uno.
Gráficos: Todo es información escrita. No hubiese quedado mal incorporar un gráfico de la marcha de la compañía.
Nivel de dificultad: Depende de la habilidad del jugador... ¡y de los «masters» en dirección de empresas!
Originalidad: Es un clásico en el terreno de los simuladores de empresas.
Conclusión: A pesar de ser un clásico, llama la atención el elevado número de «variables económicas», lo que le hace muy atractivo para quienes estén interesados en los juegos de empresas. Quizás parece haber demasiadas huelgas, pero eso depende de la gestión que cada uno realice en «su empresa».

¿FORTH en m

Para trabajar con FORTH en el Spectrum, como con cualquier otro lenguaje distinto del BASIC, es necesario un compilador. Alfonso Martínez responde a las preguntas que los lectores nos han formulado.

Los nuevos lenguajes, como el FORTH, son, en general, programas en código máquina, con algunas excepciones nada espectaculares en BASIC, que normalmente toma subrutinas de la ROM como medida de ahorro de memoria. Esto en cierta medida adultera un poco el resultado, ya que la ROM del Spectrum está pensada para funcionar como intérprete y para ser usada por principiantes, aunque en realidad con estos programas se puede hacer exactamente lo mismo que si de verdad dispusiésemos de un sistema operativo en el lenguaje que estemos trabajando.

En la mayoría de estos programas es posible encontrar dos posibilidades de uso, bien como intérprete o como compilador. Básicamente la diferencia de cada posibilidad se puede resumir viendo las propiedades de cada una de ellas.

Intérprete

El programa es interactivo con el usuario, es decir, a medida que se va introduciendo el programa o los comandos, los corrige si es necesario y los ejecutan secuencialmente. Suele ser más largo y complicado que el compilador, dadas las múltiples situaciones que debe ser capaz de salvar. En ellos es prácticamente imposible el bloqueo (siempre y cuando sean medianamente buenos). Un ejemplo muy cercano a nosotros es el ROM del Spectrum, que no es ni más ni menos que un intérprete de BASIC escrito en código máquina.

Compilador

El programa no es interactivo con el usuario, es decir, debe escribir todo el programa, ejecutar la subrutina que localiza errores sintácticos y posteriormente compila, que no es ni más ni menos que traducir el programa hecho en lenguaje de alto nivel a uno de bajo nivel como puede ser Assembler. En la mayoría de los programas comerciales se aprovecha el intérprete construyendo un programa compilado consistente únicamente en llamadas a subrutinas que se encargan de hacer lo que el programa mande. Esta opción tiene generalmente el problema de los bloqueos, ya que la rutina de corrección es tan sólo sintáctica, pero no prevé resultados absurdos que pueden bloquear el programa.

Lógicamente el FORTH no iba a ser una excepción y tiene varias versiones de programas traductores. Sólo existe un microordenador preparado exclusivamente para el FORTH (Júpiter Ace), por lo que los curiosos necesitamos de programas para hacer nuestros pinitos, que en ciertas aplicaciones es insustituible. Para Spectrum existen varias versiones, aunque algunas de ellas, escritas en BASIC en su mayoría, es preferible obviarlas. Básicamente nos vamos a referir a dos cuya disposición es parecida aunque una de ellas sea difícil de conseguir: Artic y Abersoft

Artic

En un principio fue pensada para el ZX-81, adaptada al Spectrum posteriormente. Deben existir varias versiones aunque parece difícil conseguirlas hoy día. Como nota destacada frente a la otra versión, aparte de ser más "purista", están las variables de control que ayudan mucho a la hora de programar al dar una idea del estado del sistema. Otra característica, más bien anecdótica, consiste en indicar el sistema que está utilizando, así utilizando el programa en un Spectrum, pulsando CPU obtendremos ZX-Spectrum. Otra curiosidad es la instrucción BYE que hace que salgamos del programa hacia el BASIC. En esta versión es la única forma de hacerlo.

Abersoft

Este es algo más "sucedáneo" ya que se le puede engañar y volver al BASIC utilizando algunas instrucciones erróneas, lo cual a veces ayuda a no liarse al no disponer de instrucción similar a BYE. Esta versión utiliza más direcciones de subrutinas de la ROM, es más cercano al funcionamiento del Spectrum en BASIC sin abandonar en ningún momento la filosofía FORTH. Como nota destacada está la posibilidad de definir caracteres que en las versiones iniciales de Artic no era posible.

¡Spectrum?

En resumen, cualquier versión puede servir para iniciarse en el FORTH y nuestra elección se debe basar sólo en la disponibilidad de uno u otro en cualquier momento. De hecho los ejemplos aquí descritos están hechos con la versión Abersoft, aunque en cualquier otra funcionaría.

Si ya dispone del intérprete o compilador FORTH, pasemos al siguiente paso: cargar el programa. Si la versión es original, se autoejecutará y aparecerá el típico mensaje del fabricante, año, programa y versión, aunque el orden puede variar de una versión a otra. En alguna puede aparecer además la memoria libre disponible. Además vamos a observar un mensaje al que vamos a acostumbrarnos bien pronto, que es OK, que indica que el ordenador no ha encontrado problemas y está dispuesto a seguir, esperando nuestras próximas instrucciones. Para no hacerle esperar, nada mejor que pulsar ENTER, a lo cual responderá con otro OK, indicando que no ha tenido problemas en no hacer nada, que es lo que le hemos dicho que haga, con lo que hemos descubierto que para comunicarnos con él después de haber introducido las instrucciones hay que pulsar ENTER.

En FORTH las instrucciones deben ir separadas por un espacio y escritas directamente, letra a le-





tra, así por ejemplo BORDER no se obtiene pulsando la B, sino escribiéndola letra a letra.

La siguiente pregunta que se formulará es cómo pasar de línea a línea dentro del FORTH. Para ello existe la instrucción CR, Carriage Return, que la utilizaremos para pasar a la línea siguiente. Así, pues, si escribiésemos:

1. (ENTER)

obtendríamos en la misma línea 1 OK, mientras que si ahora escribiésemos:

1. CR (ENTER)

obtendremos no en la misma línea, sino en la siguiente: 1 OK. Así se puede ver la diferencia al utilizar CR. Lógicamente se podrán escribir cuantos CR queramos mientres estén separados por espacios sin preocuparse del scroll, que en FORTH se ejecuta automáticamente.

La instrucción . lo único que hace es pintar el contenido del *stack*.

Aquí surge la diferencia entre compilador e intérprete. Lo hecho hasta ahora ha utilizado la versión intérprete donde secuencialmente se han introducido las instrucciones y se han ejecutado al pulsar ENTER. Si esto lo quisiésemos hacer en versión compilada deberíamos escribir:

: al 1 . CR ; (ENTER)

y para hacerle funcionar, deberíamos pulsar:

al (ENTER)

obteniendo lo mismo que antes aunque con una pequeña diferencia: para repetir el proceso no hay que volver a escribirlo, sino volver a pulsar a1 y ENTER.

En realidad lo que hemos hecho ha sido definir una palabra, una definición nuestra. Básicamente la diferencia respecto a lo que habíamos hecho hasta ahora reside en los distintos signos utilizados que analizamos paso a paso:

":" Al encontrarlo, el ordenador sabe que es principio de compilación, de inicio de definición, esperará un nombre.

"a1" Es el nombre que espera el ordenador, y al cual habrá que referirse cuando queramos ejecutar la instrucción. No tiene su elección mayor problema que el primer caracter debe ser una letra, no debe contener ningún espacio y su máxima longitud es de 32 caracteres. El ordenador será capaz de discernir entre vocablos diferentes, siempre que el orden, cantidad o número de caracteres de la definición sea diferente, siendo así diferentes las definiciones:

a1 y a2 o COMP y COPM o ERE2 y ER2

"1 . CR" Similar a la versión intérprete. Son sencillamente las instrucciones a realizar.

";" Al llegar a este punto, el ordenador entiende que la compilación ha terminado y vuelve al modo intérprete, dando el informe OK.

"... aproximadamente cuatro veces más rápido que el BASIC"

Si ahora pulsamos VLIST obtendremos una gran tabla de palabras predefinidas, pero curiosamente la primera en aparecer será A1, que es la que hemos definido.

Definamos ahora la palabra:

: A2 . "ESTA PALABRA NO HACE NADA" ; (ENTER)

Si pulsamos VLIST podremos verla definida al lado de A1. Si ahora escribimos:

FORGET A2

y escribimos VLIST ahora no aparece, e incluso si escribimos A2 aparcerá un mensaje O, de error: no existe esa definición, ya que la instrucción FORGET elimina la palabra especificada.

Ejemplos de compilación se pueden encontrar en el programa de números romanos que en FORTH aparece en TODOSPECTRUM en el número de enero, donde se definen las palabras, aunque hay zonas que se usa el modo intérprete como la segunda línea donde se definen las variables.

Otra diferencia apreciable entre los dos modos es su velocidad de interacción, que se puede mostrar para el siguiente ejemplo:

Pulsemos . "Hola" (ENTER), y observemos la interacción. Ahora definamos : A3 . "Hola" ; (ENTER) y pulsemos A3. ¿Nota la diferencia?

Sin embargo, dentro del modo compilador podemos introducir las instrucciones entre corchetes, como siempre separado con espacios. También tienen efecto parecido instrucciones como LITERAL o INMEDIATE.

Existen versiones de FORTH en las que se pueden escribir programas con números de línea, pero no hay que olvidar que no son números reales como puedan ser en BASIC, sino meramente representativos. En las versiones con esta opción se pulsa: EDITOR, indicando el ordenador el trabajo a realizar.

Otra posibilidad muy útil que el FORTH dispone es crear un voca-

(pasa a la pág. 24)

Exclusivo para lectores de **Todospectrum**

CLUB DE SOFTWARE

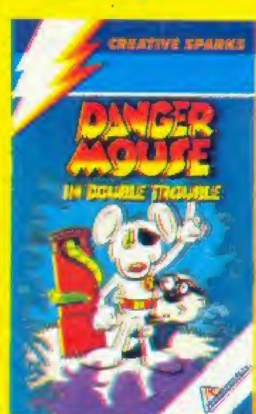
Ha seleccionado ocho excelentes programas de juego para su ordenador SPECTRUM



Precio: 1.550 ptas.

BLACK HAWK

Usted es un piloto de pruebas altamente entrenado para pilotar el avión más destructivo hasta ahora creado. Dos pantallas con casi 30 niveles de habilidad.




Precio: 1.550 ptas.

DANGER MOUSE

El malvado Barón Silas Greenback ha construido un androide que Danger Mouse debe destruir sin perder un momento.

Guíe a Danger Mouse y a Penfold a través de la espesa jungla hasta el lugar donde se encuentra el Barón, y después deténgale antes de que sea demasiado tarde.

¿Puede usted ayudar a Danger Mouse a salvar el mundo?



Precio: 1.550 ptas.

TOWER OF EVIL

¡Arriesguese!

¿Puede usted recuperar el tesoro perdido? Una acción que discurre a traves de 40 o mas habitaciones le aseguran horas de juego que cautivara su atención.

- Numero de jugadores: uno
- Control joystick o teclado
- Dos pantallas de graficos de acción rápida
- El VIC-20 requiere un cartucho 8K.



Precio: 1.550 ptas.

DELTA WING

Delta Wing es un avanzado simulacro de combate en vuelo. Ha sido diseñado para dar la excelente "sensación" de los giros y volteretas de la alta velocidad en vuelo. Con 15 controles independientes y aviones de ataque enemigos para derribar, este programa le lleva a usted y a su Spectrum al limite.

Delta Wing también permite conectar dos ordenadores Spectrum para jugar uno contra otro mediante el interface 1.



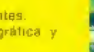
Precio: 1.550 ptas.

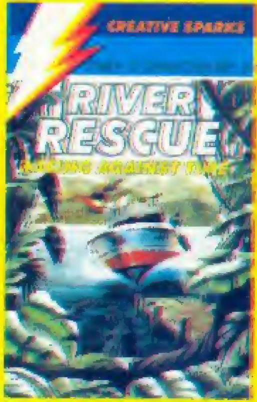
SPECIAL DELIVERY

Santa Claus se encuentra metido en un terrible lio: se durmió en los laureles, y ahora no tiene suficientes regalos para visitar a todos los impacientes niños.

Ayude a Santa Claus a recoger regalos adicionales y después a descender sobre los tejados para entregar los regalos introduciéndose por las diversas chimeneas.

- Tres pantallas rodantes.
- Amplia animación gráfica y musica.





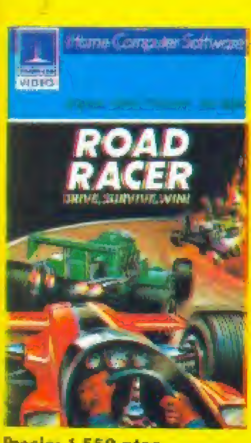
Precio: 1.550 ptas.

RIVER RESCUE

Un equipo de importantes cientificos han quedado atrapados en una peligrosa jungla, y la unica manera de sacarlos de allí es a través del rio. Usted es el propietario de una potente fueraborda y su trabajo es rescatarlos.

Navegar a través de uno de los más salvajes y tortuosos rios del mundo no resulta facil, pero usted debe dirigir su magnifica nave corriente arriba sorteando pequeñas islas, evitando cocodrilos y leños flotantes a su paso. Disfrute la rapida y excitante acción en este juego de alta energia. La acción en la pantalla rodante y el magnifico sonido y graficos le aseguran largas horas de diversion.

Recorte y envíe este cupón HOY MISMO a: **INFODIS, S.A. Bravo Murillo, 377-5.º-A. 28020 MADRID**



Precio: 1.550 ptas.

ROAD RACER

Adelanta al primer coche y a otro mas, su pulso se acelera y la tensión crece... ¿podrá tomar la delantera y ganar el trofeo?



Precio: 1.550 ptas.

GOLD RUSH

La emocionante aventura de la búsqueda de oro en un extraño planeta llamado Orón. ¿Qué peligros le aguardan...?

ENVIE A MI DOMICILIO

LA CASSETTE O CASSETTES RELACIONADAS A CONTINUACION EL IMPORTE LO ABONARE:

POR CHEQUE ☐ CONTRA REEMBOLSO ☐ CON TARJETA DE CREDITO ☐

American Express ☐ Visa ☐ Interbank ☐ Fecha de caducidad ____________

Número de mi Tarjeta ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Nombre ____________

Dirección ____________

Ciudad ____________ D.P. ______

Provincia ____________

cantidad	producto	ptas.	total

SIN MAS GASTOS DE ENVIO



(viene de la pág. 22)

bulario propio al que definimos como: VOCABULARY MIO (ENTER), donde lo que hemos definido es un vocabulario que se llama MIO y todas las definiciones que posteriormente hagamos estarán clasificadas dentro de un vocabulario definido. Si quisiésemos borrar este vocabulario por cualquier circunstancia, pulsaríamos:

FORGET MIO.

"... tan potente como el código máquina y tan sencillo como el BASIC"

En FORTH existe además la posibilidad, inexistente en BASIC, de poder conocer los valores de las varibles del sistema con sólo llamarlas, como por ejemplo:

FIRST?

apareciendo en pantalla el valor del primer *bite* de programa, cosa que en BASIC no es posible directamente. Al igual que en BASIC, en FORTH podemos cargar y grabar con instrucciones SAVE, LOAD y FLUSH.

Una vez que el programa está introducido, para hacerle funcionar hay que invocar el nombre de la definición es decir, no existe una palabra como RUN, sino que para correr cada bloque se llama por su nombre. Lógicamente, dentro de una misma instrucción se pueden introducir palabras ya definidas, como en el ejemplo siguiente:

```
: Al1 . "NUMERO" ;
: AL2 10 o do AL1 I . CR LOOP ;
```

Desde dentro de AL2 se llama a AL1.

Además existen subrutinas muy útiles en FORTH como cuando queremos introducir números con más de un caracter, pudiendo utilizar:

: INPUT PAD 1+ 64 EXPECT .O PAD (NUMBER) DROP DROP ;

esta subrutina se usa en el programa de números romanos antes citado. Si sólo queremos introducir números desde el 0 al 9 podemos utilizar:

: UNCAR KEY 48 - DUP .;

donde KEY es similaral INKEY$ del BASIC, conteniendo el código del caracter y 48 - normaliza el resultado.

También es posible trabajar en cualquier base entre 2 y 36, aunque las más usuales sean la decimal y la hexadecimal, pudiendo invocar una u otra con DECIMAL o HEX.

La versión Abersoft contiene además subrutinas típicas de la ROM del BASIC, pero con la misma filosofía del FORTH, como pueden ser PLOT, DRAW, INK, PAPER o BORDER. Ejemplos de sus usos son, comparándolas con el BASIC:

BASIC	FORTH
PLOT 33,33	33 33 PLOT
INK 7	7 INK
PAPER 2	2 PAPER, ETC.

Y si se lo toma en serio y empieza a trabajar con su ordenador, descubrirá que, al igual que otros lenguajes... ¡también obtendrá mensajes de error! Los más usuales se ofrecen en la figura 1.

Alfonso Martín

Figura 1: Informes más usuales del FORTH.

Código de error	Mensaje de error
0	*Palabra que no está en el Diccionario.*
1	*Stack* **vacío. Probar con .**
2	**Diccionario lleno. En principio difícil.**
4	**Cuidado, nombre no único. No compila el último nombre. Probar: : ALM1 . "PRUEBA" ; : ALM1 . "*" ; aparecerá el informe de error y al hacer VLIST sólo habrá un ALM1**
7	*Stack* **lleno. Igual que 2**
17	**La palabra debe ser usada en definición, al igual que en BASIC no puede usarse en programa EDIT**
18	**Ejecución sólo**
20	**Definición no acabada. Falta ; o THEN o similares**
21	**En diccionario protegido. Intentar FORGET UDG**
24	**Vocabulario protegido**

```
0 VARIABLE CONTADOR
: CUENTAS 0 CONTADOR !
CR
0 BEGIN
KEY DUP EMIT
CONTADOR @ 1 + CONTADOR !
DUP 13 = UNTIL
CR ." LA LONGITUD ES " CONTADOR @ . " CARACTERES "
DROP CR
BEGIN
EMIT DUP
WHILE
REPEAT
DROP CR ;
```



Hoy, en solitario.

Dentro de muy pocos años, la informática será un instrumento imprescindible para el conocimiento y el desarrollo de la sociedad. Y para construir este futuro, es necesario que hoy surjan, se promocionen y se ponga de relieve la labor silenciosa y en solitario de muchos nuevos valores que con toda seguridad existen.

Para hacerlo posible, la Generalitat de Catalunya organiza el 1er Festival de Software. A él pueden enviar sus trabajos todas aquellas personas o grupos que hayan creado programas informáticos.

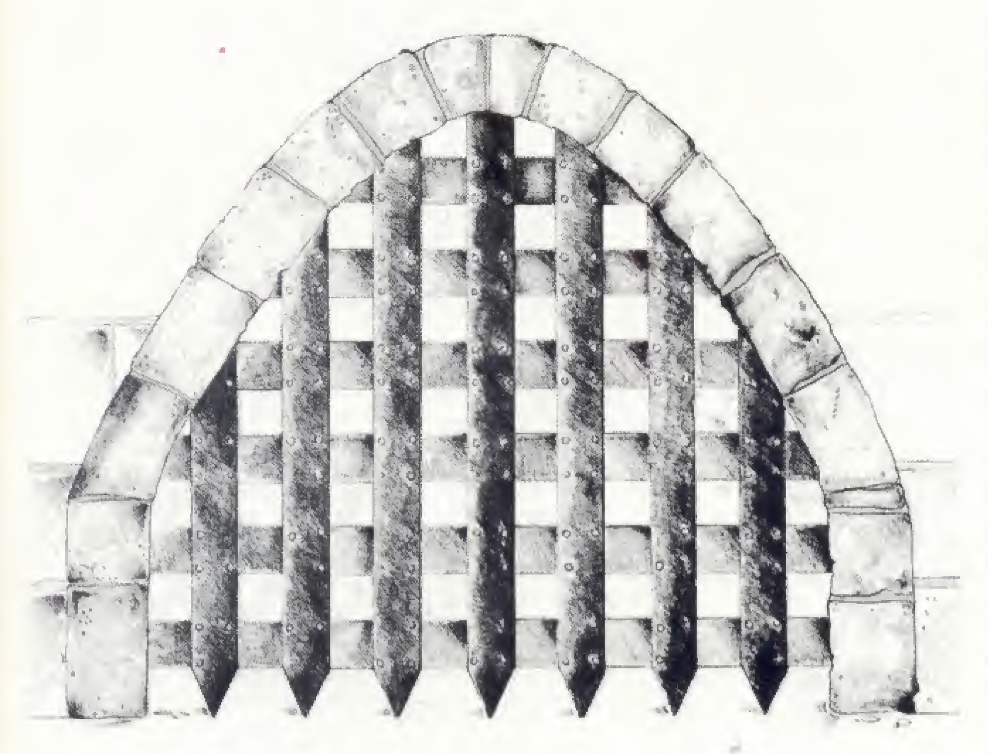
La fecha límite para la presentación de programas, será el 30 de Mayo, y en Octubre, se exhibirá públicamente los seleccionados, que serán puntuados por el público, con cuyos criterios y los del jurado se entregarán los diversos premios, que incluyen una categoría juvenil.

Participa. Y ojalá éste sea tu primer paso hacia el éxito.

MMLB



Programas / Juegos / Montajes / Còdigo Màquina
AÑO II - Núm. 17 - Abril 1985 - 250 Ptas.
ZX
REVISTA PARA LOS USUARIOS DE ORDENADORES SINCLAIR
Mapa para jugar al ATIC-ATAC
Física: Dinámica de una partícula
Estiiiiira tus caracteres
Un día en las carreras
YA ESTA A LA VENTA
SUPLEMENTO
¡YA ESTA AQUI!
QL
Ultimas novedades
¡GANE ESTA MOTO!



Una clave, please

El almacenamiento de información en *cassette* es muy eficiente, pero el método elegido para estructurarlo hace que la protección de los programas sea difícil. Cada bloque de datos o programas está precedido por una cabecera, que es un pequeño bloque de datos conteniendo información sobre las características principales, por ejemplo, la longitud, dirección de comienzo y el tipo de datos que hay.

Los usuarios que hacen su propio *software* no tienen forma de proteger sus programas para evitar su lectura ilegal, por lo que puede ser útil una rutina que requiere la introducción de una clave para poder entrar al listado. La rutina no permitirá el acceso al programa hasta no introducir la clave correcta.

La clave podrá tener la longitud que especifiquemos, siempre y cuando tenga al menos un caracter, y se puede alterar con la información adecuada, siempre que haga falta. Una vez aclarado esto, y programado, la rutina hace un buen trabajo al evitar que se abran los programas.

También hay una característica incluida para las distintas versiones que se han de hacer. Por razones de seguridad el listado desensamblado del código máquina no se ha incluido, pero más adelante se explica detalladamente. El código, que está en el programa como un volcado decimal, no es relocalizable. Sin embargo, para prevenir que se limite la utilidad del código, el programa en BASIC requerirá que se le introduzcan los detalles de la asignación de memoria, y según se vuelca el código a la memoria se ajusta la dirección absoluta para que corresponda con la dirección requerida. Esto significa que cada vez que quiera preparar la rutina se ha de ejecutar el programa cargador de nuevo. Si se desea añadir la rutina a un programa extenso que ocupe la mayor parte de la memoria, será necesario situar la rutina en el *buffer* de la impresora, que será posible gracias a que sólo ocupa 229 bytes de longitud.

El método de introducir la clave y la rutina de comprobación se han diseñado cuidadosamente. Cuando llamemos a la rutina la pantalla se limpiará y aparecerá la pregunta "INTRODUZCA LA CLAVE". A continuación se ha de introducir la clave correcta.

Deberá anotar los datos siguientes: mientras se introduce la clave se escuchará un *beep* después de cada pulsación; cualquier cosa que se introduzca no aparecerá en la pantalla; no se ha incluido la función de autorrepetición de las teclas y al finalizar de introducir la clave se ha de pulsar ENTER.

La clave no se visualizará en la pantalla mientras se está tecleando, de manera que si alguien está observando cuando se introduzca, encontrará difícil descifrar lo que se está tecleando. Es fácil seguir mentalmente dónde se encuentra uno mientras se introducen los datos, especialmente con el *beep* del teclado. No existe la posibilidad de autorrepetición y esto le evita equivocarse debido a dejar una tecla pulsada más del tiempo deseado.

Aunque la necesidad de pulsar ENTER después de introducir la clave parece ser innecesaria, y se podía evitar, tiene una serie de ventajas. Permite la posibilidad de utilizar claves de distinta longitud, en vez de mantener la misma longitud siempre, ya que esto sería una limitación, además permite a la rutina llevar un control acerca de los intentos llevados a cabo al introducir la clave (la importancia de esto se verá más adelante) y también evita que cualquier usuario que desconozca la clave dé con

ella por equivocación y se introduzca en nuestro programa han de pulsar ENTER para que se compare la palabra introducida por la clave.

Si ya ha visto el programa cargador, observará que a la variable "offset" ha de asignársele un valor cuando se ejecuta el programa. Esto es una característica de la rutina que permitirá efectuar versiones alternativas de ella, cada una algo distinta de otra. A la variable "offset" se le ha de asignar un valor entre 0 y 128 inclusive. No es necesario acordarse de dicho valor cada vez que introduzca la rutina, pero dando valores diferentes aseguraremos que la rutina sea distinta cada vez que accedemos a ella, haciendo más difícil "romper" los programas.

No se permite el acceso al programa hasta que se introduce la clave correcta

Cuando tecleamos la clave deseada en el programa cargador, se debe tener en cuenta que las mayúsculas y minúsculas se diferencian y cuando el código máquina lea del teclado, sólo se verán letras minúsculas, a no ser que se mantenga pulsada la tecla CAPS SHIFT mientras se teclea. La rutina aceptará números e incluso cualquier caracter que se obtenga al pulsar la tecla SYMBOL SHIFT excepto los comandos (NOT, STEP, TO, AND, ETC). Si utilizas comandos dentro de la clave, la rutina se vendrá abajo, o en el mejor de los casos será imposible de introducir.

Tampoco podrá utilizar ningún caracter que se obtenga con el cursor en el modo "E", por ejemplo, no utilizar cualquier caracter o comando que estén en la parte superior e inferior de las teclas en rojo y verde respectivamente, ya que estos son imposibles de introducir en código máquina. Para evitar desastres, se deberán probar las claves antes de incorporar la rutina a nuestros programas. Es mejor asegurarse que la clave funciona correctamente antes de que nos sea imposible introducirnos en nuestro propio programa debido a que hemos elegido una clave ilegal.

Otra característica de esta rutina que excluye a los "piratas" está el tiempo de retardo que empleamos conjuntamente con el teclado. Está implementado de dos maneras distintas: la primera es como un dispositivo de autodestrucción. Mientras se ejecuta la rutina el programa cuenta el tiempo que se tarda entre pulsación y pulsación. Si alguna vez este tiempo supera los diez segundos el ordenador se borrará (NEW) automáticamente. Esta característica está diseñada para evitar que se prueben las combinaciones posibles secuencialmente.

La otra característica también ayuda al respecto, aunque no cuenta en tiempo real. La rutina contará el número de intentos llevados a cabo. Podrá equivocarse una vez, incluso dos veces, pero a la tercera se autodestruye, borrando todo.

Añadir la rutina en código máquina a los programas ya escritos no es ningún problema. Puede llamar a la rutina desde el inicio del código máquina mediante una llamada (CALL) a la dirección de comienzo y luego proceder a la ejecución del programa. Si el usuario se equivoca al introducir la clave, será imposible acceder a la rutina principal, en código máquina.

También es posible añadir la rutina a programas en código máquina, con una condición: tiene que haber la suficiente memoria libre para poder ubicar la rutina, la clave y el *buffer* de entrada de datos. Si encuentra ese espacio suficiente, la conversión es inmediata. Todos los programas en código máquina han de ser llamados desde un pequeño programa en BASIC. Este programa puede constar de dos instrucciones:

1 LOAD "" CODE: RANDOMIZE USR dirección de comienzo

Esto permitirá, que al autoejecutar un programa se cargue la rutina y llame al código máquina. El programa cargador podrá contener comandos (INK, PAPER, BORDER, ETC) que limpiarán la pantalla y la dejará presentable para cargar el programa, hasta podrá tener más de un bloque para cargar. Esto no le afectará.

Rutina no relocalizable de 229 bytes

Para incluir la rutina de seguridad de la clave haga lo siguiente:

1. Modifique el programa cargador al comienzo del *software* de la manera siguiente: después de todos los comandos LOAD "" CODE, inserte dos instrucciones LOAD "" CODE más.
2. Escriba la dirección de comienzo de RANDOMIZE USR dirección de comienzo, normalmente al final (esto puede ser alternativamente LET a = USR dirección de comienzo o también un PRINT USR dirección de comienzo.
3. Trabaje donde coloca los 229 de la rutina en la memoria y escriba esta dirección en la instrucción RANDOMIZE USR en el programa, de manera que el BASIC llame a la rutina de seguridad.
4. Guarde (SAVE) el programa cargador en BASIC modificado encima del programa antiguo al comienzo del *software.*
5. Carque (LOAD) o teclee la rutina cargadora de la clave.
6. Convierta la dirección que escribió anteriormente en el formato HOB/LOB *(High/Low order byte),* tal y como aparece en la página 173 del manual, con la excepción de que no se hace POKE n, (o POKE n+1), sólo se imprime.
7. EDITe la última línea de la DATA en el programa cargador, sustituya el 201 y en su lugar pondremos, primero 195, luego una

(pasa a la pág. 39)


QL
MAGAZINE
Suplemento especial
Abril 1985
¡YA ESTA AQUI!
Ultimas novedades
Análisis de software
Lenguajes: SuperBasic, Pascal, Assembler, Lisp.
Entrevista
Aplicaciones made in Spain
ANDRÉS SANCHEZ

¡Por fin llega el QL!

Pocos ordenadores han levantado tanta expectación como el nuevo Quantum Leap, conocido simplemente como QL. Y es que, hay que reconocerlo, Sinclair sigue siendo el líder mundial en ordenadores de bajo precio y seguirá siéndolo por mucho tiempo mientras continúe batiendo sus propios récords, desde el ya legendario Timex 1000 que no llegó a comercializarse en España, hasta el nuevo QL, dispuesto a pisar fuerte en el mercado de ordenadores doméstico-profesionales.

¡Por fin llega el QL!

Pero no llega solo. En este suplemento analizamos los distintos productos hard y soft que lo acompañan, junto a los proyectos españoles en marcha y que pretende ser el embrión de una futura revista independiente de usuarios QL.

Muchas son las noticias que hemos de daros sobre el mundo Sinclair, por el momento casi exclusivamente del mercado inglés.

En esta sección de noticias, tan solo nos fijaremos en los más importantes, presumiblemente aquellas que podremos tener en España en breve.

Monitores

La alta resolución del QL, junto con la anchura de la imagen que produce han motivado la aparición de numerosos monitores "especiales QL".

Los formatos elegidos son 12 y 14 pulgadas en color a precios asequibles, a partir de 96.000 ptas. el modelo de **Microvitec**, primero en aparecer en el mercado español.





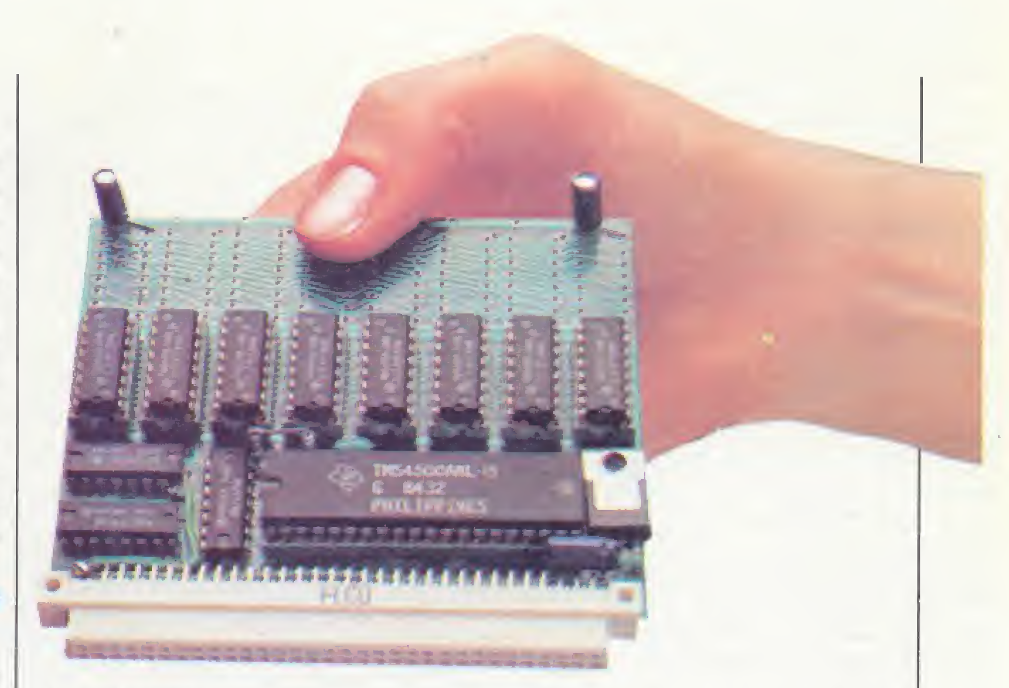
Libros

Se prepara una gran avalancha de información escrita sobre todo lo que usted pueda imaginar sobre este ordenador. La editorial **RA-MA** parece ir por delante con sus dos primeros títulos ya a la venta, en español: "Manual de referencia para el Sinclair QL" de Tim Hartnell, y "Explorando el Sinclair QL. Una introducción al Superbasic" de Andrew Nelson.

CP/M para el QL

La empresa **QL+** comercializa una tarjeta con un Z80, 64K de memoria, y el *software* necesario para la comunicación con el QL. En pocas palabras, una tarjeta CP/M. La extensión incluye un *interface Centronics* y nos da la posibilidad de usar la gran cantidad de programas que existen para ese operativo.

Por otro lado **Quest** ha desarrollado una versión del CP/M que "corre" con el procesador 68008, pensando en su uso en máquinas desarrolladas a partir de este procesador. El sistema funciona con *microdrives* o *diskettes* y se plantea como una alternativa al QDOS.



Esto es el K/OS

El 68 K/OS es un sistema operativo, resultado de un contrato de GST con Sinclair para diseñar el sistema operativo del QL. Finalmente no se montó con la máquina y ahora la empresa lo ofrece en ROM, como una alternativa al QDOS. El sistema ofrece una serie de utilidades en Microdrive, pero (de momento) sólo se puede programar con el ensamblador, por lo que parece difícil que logre ocupar el sitio del QDOS.

¿Qué QL?

Existen en este momento varias versiones del sistema operativo del QL. Cuanto más moderno, menos errores. Para saber la "edad" de nuestra máquina, basta teclear PRINT VER$. El ordenador escribirá FB, AH o JM. Esta última corresponde a la versión vigente en este momento. La versión 2.0 que se lanza en breve parece que irá etiquetada como JS.

Interfaces

Ante la falta de *interface centronics* standard para el QL muchas empresas se han lanzado a "tapar el hueco", por lo que en el mercado inglés abundan los adaptadores e *interfaces* para conectar impresoras *Centronics* al QL. Los precios están alrededor de las 8.000 ptas.

Para quienes quieran utilizar su QL en el control de instrumentos de laboratorio u otro tipo de aparatos que utilicen el standard IEEE-488, **CST** manufactura un *interface* para utilizar este standard, también utilizado por impresoras y *plotters* de calidad. El puerto de comunicaciones es totalmente accesible desde el QDOS.





Discos

Hasta el momento ya han salido cuatro versiones de controladores de disco flexibles (hasta 720 Kbytes) y una unidad de disco duro de 7.5 Megabytes.

Quest, una de las casas de *software* especializada en el QL, "obliga" a utilizar CP/M 68K (Sistema operativo diseñado para el QL por la misma firma) para utilizar sus unidades de disco. Las restantes unidades pueden funcionar con el QDOS (Sistema operativo del QL).

En cuanto a precios, hay gran variedad dependiendo del *interface* y tipo de disco elegido, oscilando entre 60.000 y las 240.000 ptas. del disco duro.

Con un diseño muy agradable a la vista, al sacar el QL de su caja nos da la impresión de que Sinclair ha progresado mucho desde el Spectrum. El teclado es "standard" QWERTY de 65 teclas, muy estético, pero no llega a la calidad profesional. Las teclas tienen un recorrido largo y poca tolerancia respecto al ángulo de pulsación. Adecuado para mecanógrafos "amateurs".

En la versión española, los acentos y la Ñ ocupan su lugar habitual en las máquinas de escribir. La tecla CONTROL, clásica de otros ordenadores, da acceso a todo el juego de caracteres. Cinco teclas de función alineadas en dirección vertical en la parte izquierda del teclado facilitan el manejo de opciones en los programas.

Si queremos conectar el QL a una pantalla, disponemos de toma de televisión (UHF) parecida a la del Spectrum. También se puede conectar a monitor, mediante una clavija RGB. El cable de TV está incluido con el aparato; no así el cable de video. La resolución gráfica es generosa, con 512 por 256 puntos en 4 colores o 256 × 256 puntos en 8 colores. La imagen tiende a "rebosar" en algunos monitores, por lo que, si ya tenemos uno, deberemos asegurarnos de que la pantalla del QL "cabe" en él.

Las características gráficas del QL ocupan 32K de las 128K que trae. Esta capacidad de memoria, ampliable a 640K, puede parecer excesiva al usuario acostumbrado a las 48K del Spectrum, pero se queda pequeña con cierta rapidez en un uso profesional, 48K de ROM incluyen el sistema operativo (QDOS) y el BASIC; sus características son comentadas posteriormente.



La parte trasera del aparato incluye conectores para periféricos. Una expansión de ROM de 16K permite extensiones del BASIC y conexión de otros lenguajes o periféricos. A continuación nos encontramos dos entradas para *joysticks* y dos "puertos" RS232C para impresoras o "modem"; los cuatro están dotados de clavijas no "standard". La conexión UHF y la salida video vienen a continuación. La lista finaliza con dos conexiones para Red Local, análogas a las del ZX *Interface I* sin olvidar la ampliación para *Microdrives* (hasta 6 más) en el lateral derecho. El *bus* de expansión, para ampliación de memoria y periféricos, queda oculto en el lado izquierdo.

En una máquina pensada para uso profesional no puede faltar un medio de almacenamiento externo. Sinclair, fiel a los *Microdrives*, ha colocado dos en la parte anterior derecha del QL. Cada uno almacena algo más de 100K, con tiempos de acceso algo peores que las unidades de *diskettes*. Su fiabilidad ha sido puesta en entredicho, aunque nosotros no encontramos grandes problemas en su uso. Tras la bajada de precio de los cartuchos su único inconveniente serio es la capacidad.

En el interior del QL encontramos un **Motorola** 68008. Este procesador es el "hermano pequeño" de la familia 68000. Sus registros son de 32 bits, y direcciona 1 Mbyte de memoria. Con un *bus* de datos de 8 bits, hay que esperar cuatro ciclos de memoria para "llenar" un registro; resulta, por tanto, algo más lento que sus hermanos. Su reloj interno trabaja a 7 MHz. Para descargar de trabajo al 68008 en las tareas pesadas hay un microordenador, el 8049 de **Intel**, que se encarga de la lectura del teclado, el altavoz y la comunicación serie.

Aunque Sinclair intenta entrar en el mercado profesional con esta máquina, no parece fácil que consiga vender en el mercado de las grandes empresas, debido a la alta sensibilidad que muestran estas casas hacia la compatibilidad **IBM**. El QL podría encajar mejor en el campo de las profesiones liberales, así como en las pequeñas empresas, donde el coste del ordenador y del *software* se convierte en un condicionante importante. Para este tipo de usuarios, el QL ofrece una excelente relación precio-prestaciones, incluyendo con la máquina unos paquetes de *software* que valen, a precios de mercado, más que la propia máquina.



Prototipo del teclado QL en español.



Para facilitarnos el "arranque" con el QL, el precio incluye cuatro programas de gestión: un procesador de texto (QUILL), una base de datos (ARCHIVE), una hoja de cálculo (ABACUS) y un programa de gráficos profesionales (EASEL).

La principal característica de QUILL es que presenta en cada momento en pantalla lo que aparecerá en la impresora. Los menús de definición de márgenes y tipo de letra son flexibles, siendo también cómoda la inserción y el borrado. El programa presenta opciones para cargar, grabar y combinar ficheros desde *microdrive*, así como la opción *import*, que permite editar textos generados por los otros programas. La incomprensible falta de un programa de *mailing*, que permitiría la personalización de cartas mediante la base de datos, es su principal defecto.

Y ya que estamos con ARCHIVE, sigamos. Es un programa muy versátil, donde el usuario dispone de un lenguaje de programación que permite la gestión de la base de datos "a la medida". Similar al SuperBasic, admite subrutinas con parámetros, variables locales y facilita la programación estructurada. La presentación en pantalla es definible mediante un editor de gran simplicidad.

Una característica muy interesante de ARCHIVE es que se puede crear un fichero sin definir "a priori" la longitud máxima de cada campo: el programa almacena registros de longitud variable. Una característica útil, ya que en un medio de poca capacidad como el *microdrive* el desperdicio de memoria se paga muy caro. Existe la posibilidad de trabajar simultáneamente con varios ficheros, ordenarlos por varios campos... En fin, sólo se le puede objetar la lentitud de acceso y la escasa capacidad de los *microdrives*.

Una imagen vale más que 1K (1024 palabras). Y lo que hace EASEL es convertir nuestros datos en imágenes. Resulta con mucho el más fácil de usar del paquete, ya que la pantalla refleja inmediatamente todas nuestras órdenes y ajusta (si es necesario) la escala. Las opciones se eligen, por menú, con valores por defecto en casi todas las situaciones.

Se puede dibujar con tres formatos básicos: diagrama de barras, líneas o "tarta". Dentro de cada uno hay una flexibilidad en la elección de formas y colores. Si queremos representar varias figuras, el programa "hará sitio" automáticamente. Se pueden introducir comentarios en el dibujo.

La capacidad es suficiente para cualquier uso realista y existe la posibilidad de "traer" los datos de ARCHIVE o ABACUS. Para copias en papel se requiere una impresora compatible Epson.

ABACUS completa el lote de programas de **Psion**. Se trata de una hoja de cálculo "inteligente". Muy útil siempre que tengamos que procesar datos, aunque su facilidad de programación hace que se usen cada vez más en todo tipo de problemas. ABACUS permite la referencia a filas o columnas mediante nombres simbólicos. También se puede cambiar a voluntad el ancho de cada columna y recorrer la tabla mediante ventanas. Su principal defecto es la escasez de memoria (remediable con la ampliación) y la falta de protección de las fórmulas. Por lo demás, un programa impecable, donde la facilidad de uso se auna con la potencia.

Pese a las duras críticas que recibieron en Inglaterra las primeras versiones de los programas, la casa ha realizado un considerable esfuerzo de mejora. En la versión 2.01 los programas residen íntegramente en memoria, y sólo se accede a *microdrive* cuando pedimos "ayuda". Están escritos en castellano y admiten los caracteres españoles. Les pudimos echar una ojeada y nos sorprendieron los tiempos de carga y la velocidad de ejecución. La guerra de precios en el mercado profesional español va a comenzar. Los próximos meses dirán si la apuesta de Sinclair le convierte de nuevo en ganador.



Siendo el QL un ordenador muy poco convencional, no cabe la menor duda de que su BASIC tenía que ser poco corriente. En efecto, el Super-Basic se parece tan poco al BASIC que incorporan otros ordenadores, que Sinclair estuvo tentado de cambiarle el nombre. Pero ésto supondría un choque psicológico para el usuario que sabe programar en BASIC, posiblemente reticente a aprender un idioma nuevo. Y en muchos aspectos el Super-Basic es un idioma nuevo.

Incorpora todas las características standard de los dialectos BASIC más usuales. Pero va mucho más allá. Por ejemplo, los comandos GO TO y GO SUB han sido incorporados "por compatibilidad" y se nos recomienda que los utilicemos lo menos posible. En su lugar, se nos recomienda el uso de procedimientos y funciones.

Los procedimientos son el equivalente a las subrutinas de otros idiomas. Para los poco familiarizados con el concepto, suponen un programa que acepta una serie de parámetros y modifica unas variables o realiza una acción. En suma, lo mismo que los comandos del BASIC, pero con capacidad para realizar acciones complejas.

Las funciones son análogas a los procedimientos, pero devuelven, como las del BASIC, un resultado. Ya no están, como en el Spectrum, restringidas a una sola línea. Su uso permite el diseño modular de programas, haciéndolos mucho más legibles.

Todos los identificadores pueden tener hasta 255 caracteres, comenzando por una letra. Esto vale para los nombres de variable, subrutina o función. Análogamente a otros dialectos, si el nombre de una variable acaba en "%" la variable es entera, lo que permite ahorrar memoria; si acaba en "$", se trata de cadenas de caracteres.

La definición de procedimientos permite ampliar el lenguaje

Otros detalles que expresan la potencia del Super Basic son las estructuras de control: la instrucción IF ... THEN queda completada por la posibilidad ELSE ... ENDIF de tomar una decisión si no se cumple la condición. Al bucle FOR ... NEXT se le añaden las palabras EXIT y END FOR, que permiten una mayor complejidad. Una construcción nueva es el bucle REPEAT ... NEXT ... END REPEAT, que, mediante la posibilidad del EXIT permite construir ciclos sin necesidad de referenciar nunca números de línea.

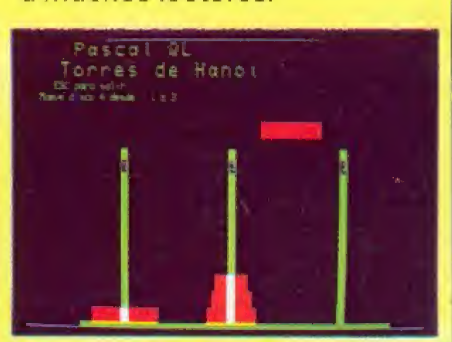
En el Super Basic no existen las funciones VAL, VAL$ y STR$. No son necesarias: la conversión entre números y cadenas es realizada automáticamente por el intérprete. Por ejemplo, a = "2" + "3" devuelve 5 en a, a$ = "2" + "3" deja a$ = "5" y a$ = "2"&3 deja "23" en a$. Un concepto potente, llamado coerción.

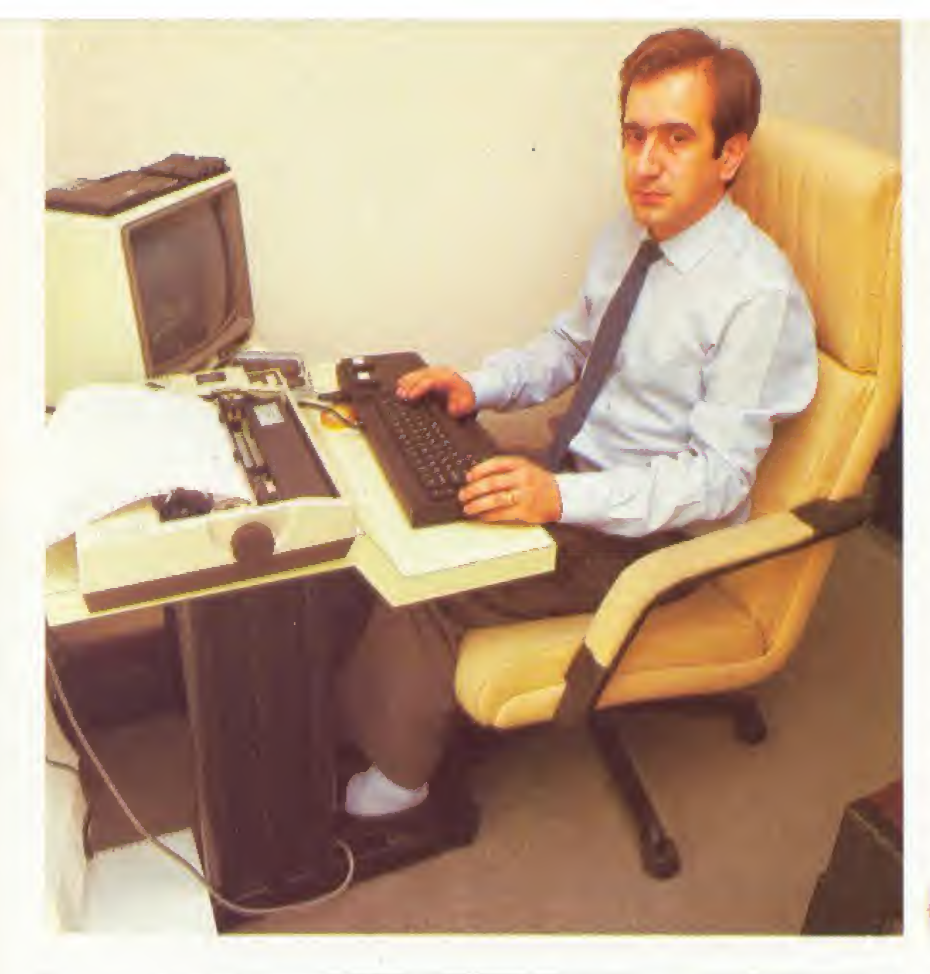
La unión de un lenguaje altamente estructurado, de potentes sentencias de control, con una gran cantidad de comandos que explotan todas las posibilidades de la máquina, más su fácil extensibilidad mediante código máquina, hacen que este lenguaje merezca efectivamente el prefijo de Super con que sus creadores lo han calificado. Uno de los puntos más fuertes de esta máquina, como candidata a su uso para el aprendizaje.

Otros lenguajes

Los compiladores e intérpretes de otros lenguajes para el QL han aparecido con una rapidez inesperada. Hasta el momento se pueden encontrar tres ensambladores distintos, dos compiladores de Pascal, un Forth... Incluso lenguajes difíciles de encontrar para microordenadores, como el Lisp, el APL, el BCPL o el C, han aparecido rápidamente para el QL. La razón es simple, se trata de una de las primeras máquinas de precio asequible que llevan un procesador de la familia 68000, a la que todos auguran un gran futuro. Y las casas de *software* se han lanzado a producir para estar instalados cómodamente cuando se amplíe el mercado. Lo que, de rebote, redunda en beneficio del usuario que quiera programar en estos idiomas. En futuros números os contaremos más acerca de estos lenguajes, que sonarán "a chino" a muchos lectores.







Hasta ahora, en este corto suplemento QL, hemos visto las características principales del ordenador. Pero nos habíamos dejado lo más importante: su buena aceptación por el público y casas de *software made in Spain*. Este es el caso de Pedro Pérez, consejero delegado de **World-Micro**, quien nos da sus impresiones, después de un largo año de trabajo con el QL: "**Investrónica** ya no es aquella casa que posiblemente por casualidad trajo el ZX81, ahora es una empresa seria y esto da confianza, sobre todo si es un producto Sinclair. Pero lo más importante es que a la gente le gusta el ordenador. Es cierto que cuesta mucho introducir al cliente, pero le sientas frente a la máquina y ve lo que puede hacer y a qué precio, después no quiere otro."

Pocos son los que conocen que QL es el nuevo ordenador Sinclair, pero poco más saben de él, como indica Pedro. "Vienen diciendo que quieren ver esa máquina de dos 'cacharritos' pequeños. Hay que ser serios, esto es un ordenador que tiene un soporte de información de 200 K al precio aproximado de 130.000 pesetas, cuando los equipos de la competencia de prestaciones similares, están por encima de las 220.000 pesetas."

Sin embargo, no sólo llama la atención su precio, sino especialmente los programas de aplicación de **Psion**, lo que denota un interés por parte de un público más profesional: "Valoran mucho los cuatro programas. La mayoría de nuestros clientes son pequeños empresarios y no están interesados en programar, les falta el tiempo o la afición. En estos casos adaptamos el Archive a sus necesidades, pero no tenemos previsto hacer programas a medida."

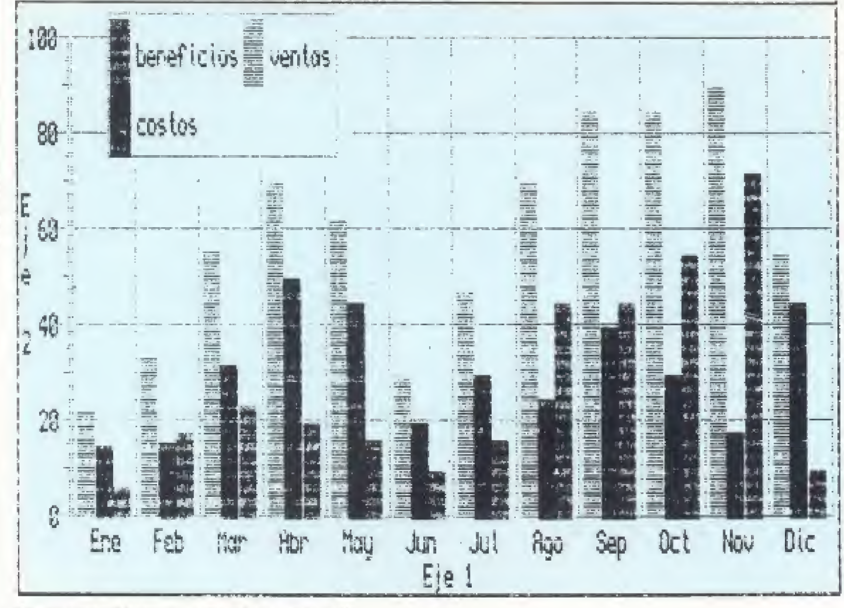
Aplicaciones made in Spain

Las casas de software no ven al QL como un ordenador de juegos, y piensan seriamente en las aplicaciones.

Investrónica ya está preparando programas de nóminas, contabilidad, agenda, paquete de arquitectura (mediciones y presupuestos, cálculo de pórticos y de estructuras) y diversas aplicaciones "verticales" para médicos, abogados, notarios, etcétera.

Alsi comercial está adoptando sus programas Spectrum para la facturación, control stocks, mailing, letras de cambio, y hasta un programa especial para dentistas.

Son tan sólo dos ejemplos de lo que nos espera.



Easel: aplicaciones "en serio".

Este dossier ha sido realizado por Santiago Gala, con la colaboración de Juan Arencibia y Miguel Gómez.

ESPECIFICAMENTE PARA QL



ESTAMOS EN INFORMAT
PALACIO FERIAL - NIVEL 2
STAND - 202

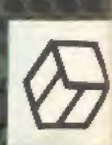
ESPECIFICACIONES

MODELO:
CUB 1451/DQ3
14" QL MONITOR
Entrada RGB-TTL
Resolución (PIXELS)
653 (H) × 585 (V)
DOT PITCH 0,43 mm
Bandwidth 18 MHz
Específicamente diseñado para el QL que aprovecha su facilidad única de 85 columnas.

Mod. 1451/DQ3
P.V.P. **96.000** Ptas.

MICROVITEC 653
cub
COLOUR DISPLAYS

DISTRIBUIDO EN EXCLUSIVA POR: MULTILOGIC P.° de la Habana, 145. 28036-MADRID Tel. 458 74 75

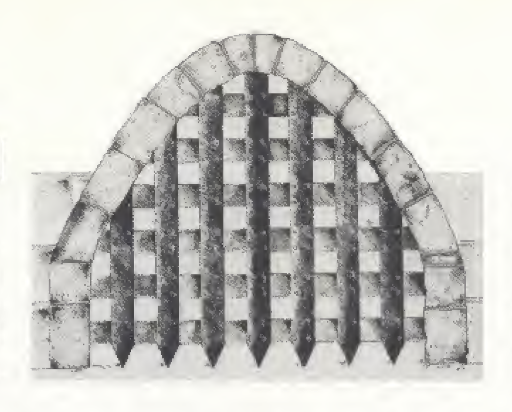


(viene de la pág. 30)

coma, a continuación el valor de LOB que se acaba de calcular, otra coma, a continuación el valor de HOB y por último una coma. Al final de la sentencia DATA todavía tiene que haber un 999.

8. Ejecute el programa (RUN), tecleando la misma dirección de comienzo que la que se puso en el punto anterior y con la clave deseada. Los bytes del código máquina estarán presentes en la memoria.

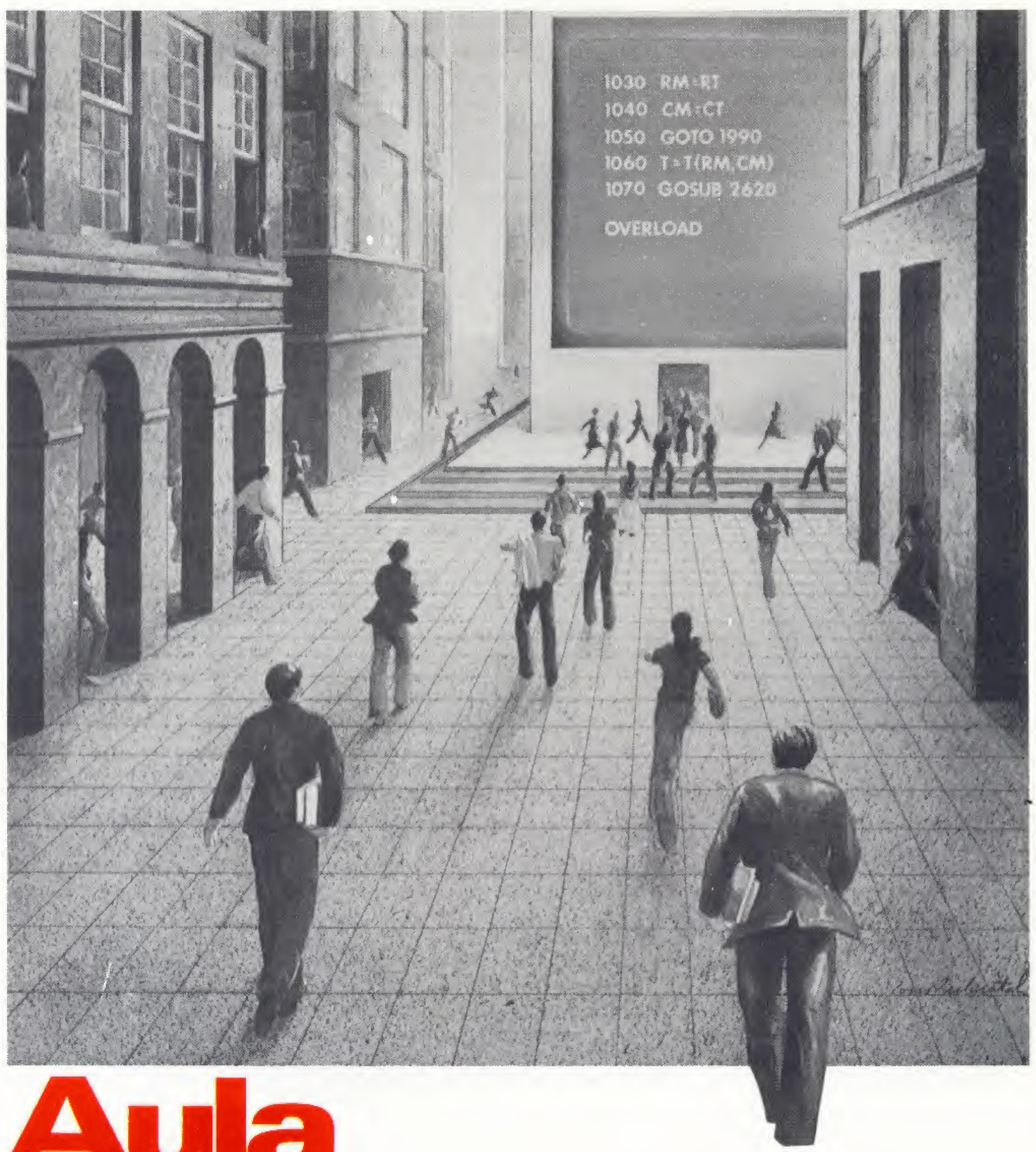
9. Guarde (SAVE) estos 229 bytes como CODE, y también la clave como CODE justo al final de la cinta con el programa, después del último bloque del programa. Observe que el programa cargador se cambió para que pudiera cargar los bytes de la rutina, la clave y la ejecute.



Cuando todo se ejecute, el programa cargador en BASIC, cargará todos los bloques de datos, incluyendo la rutina, la clave y antes que nada llama a la rutina de seguridad, que pasará el control al programa principal una vez introducida la clave correcta. La rutina no vuelve al BASIC porque la hemos cambiado para que ejecute directamente el programa que se encuentra en la memoria.

Finalmente, evite utilizar claves que le sean familiares al ser éstas demasiado obvias: distritos postales, nombre de animales, números de teléfonos, etc.

```
   1 POKE 23609,15
   5 INPUT "Donde situo el RAMTO
P? ";rt
  10 CLEAR rt
  20 DIM h(6): DIM l(6)
  30 INPUT "Direccion del codigo
 (longitud  = 227 bytes): ";a
  40 INPUT "Valor del offset: ";
v
  50 LET v=INT v: IF v<0 OR v>12
8 THEN  GO TO 40
  60 LET offset=v
  70 INPUT "Direccion del buffer
 de entrada ";b
  80 INPUT "Direccion de la clav
e: ";p
  90 LET h(1)=FN h(b): LET l(1)=
FN l(b)
 100 LET h(2)=FN h(p): LET l(2)=
FN l(p)
 110 LET h(3)=FN h(a+6): LET l(3
)=FN l(a+6)
 120 LET h(4)=FN h(a+42): LET l(
4)=FN l(a+42)
 130 LET h(5)=FN h(a+50): LET l(
5)=FN l(a+50)
 140 LET h(6)=FN h(a+58): LET l(
6)=FN l(a+58)
 150 INPUT "Clave? ";p$: IF p$="
" THEN  GO TO 150
 160 LET p$=p$+CHR$ 13: FOR f=1
TO LEN p$: POKE f+p-1,offset+(CO
DE p$(f)): NEXT f
 190 CLS : PRINT AT 10,7;"Volcan
do el codigo"
 200 DATA 175,50,129,92,24,52,22
,11,3,61,61,61,61,61,61,61,61,61
 210 DATA 61,61,61,61,61,22,10,3
,69,78,84,69,82,32,80,65,83,83
 220 DATA 87,79,82,68,58,32,82,6
9,74,69,67,84,69,68,65,67,67,69
 230 DATA 80,84,69,68,58,129,92,
60,50,129,92,254,4,32,1,199,58
 240 DATA 4,92,60,32,250,205,107
,13,33,144,1,17,144,1,17,144,1
 250 DATA 205,181,3,62,2,205,1,2
2,17,l(3),h(3),1,36,0,205,60,32
 260 DATA 33,0,0,1,l(1),h(1),33,
120,92,54,0,35,54,0,62,2,190,32
 270 DATA 1,199,58,4,92,60,40,24
4,197,33,151,1,17,40,0,205,181
 280 DATA 3,193,58,8,92,198,offs
et,2,3,254,13+offset,40,8,58,4
 290 DATA 92,60,40,208,24,248,17
,l(1),h(1),33,l(2),h(2),26,190
 300 DATA 32,9,62,13+offset,190,
40,17,19,35,24,243,17,l(4),h(4)
 310 DATA 1,8,0,205,60,32,14,1,2
4,11,17,l(5),h(5),1,8,0,205,60
 320 DATA 32,14,0,6,8,33,83,89,1
26,246,128,119,35,16,249,6,255
 330 DATA 118,16,253,12,13,194,l
(6),h(6),205,107,13,201,999
 340 RESTORE 200
 350 READ z: IF z<>999 THEN  POK
E a,z: LET a=a+1: GO TO 350
 360 DEF FN h(u)=INT (u/256)
 370 DEF FN l(u)=u-256*FN h(u)
```



Aula informática con Spectrum

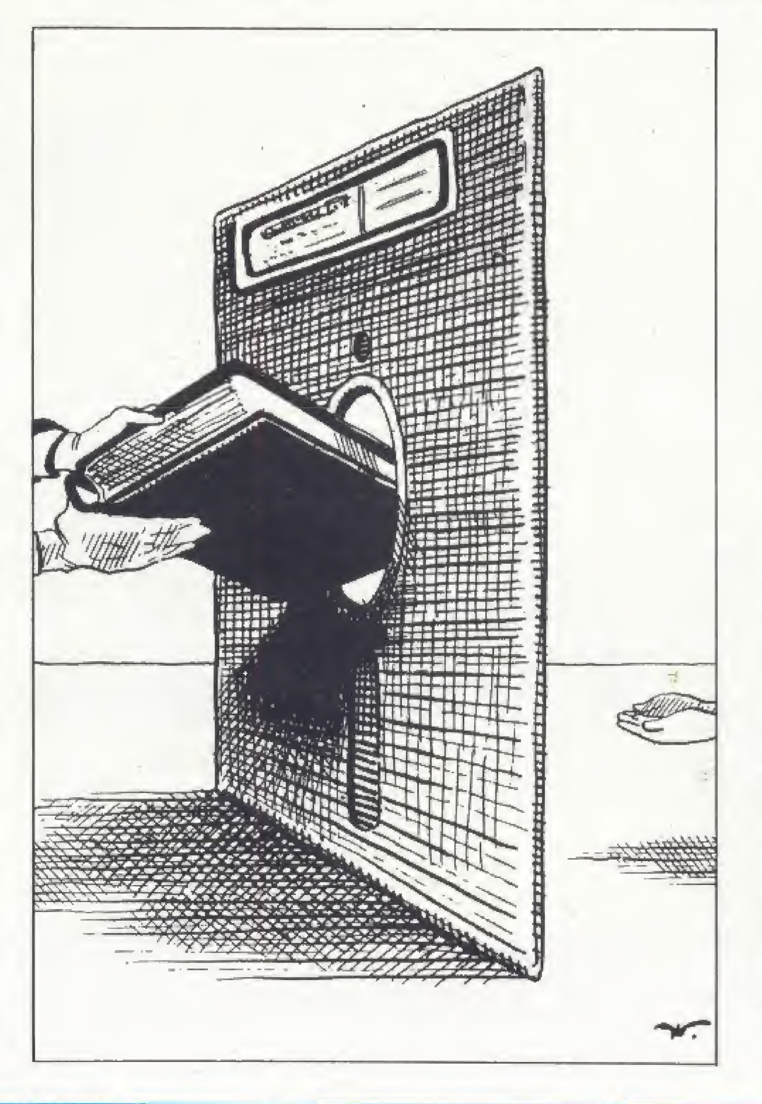
Día a día crece la preocupación por la introducción de la informática en las escuelas, sin que por el momento exista una definición clara del ministerio. Ello redunda en la elección de equipos poco adecuados, en algunos casos, y en una deficiente utilización de los mismos, en la mayoría de los casos. Miguel Figini, experto en aplicaciones informáticas a la enseñanza, nos da su visión sobre el tema.

La progresiva introducción de la informática en los centros escolares plantea toda una serie de nuevas cuestiones, tanto a los educadores como a los profesionales de la computación. Nos proponemos abordar con esta serie de artículos distintos aspectos que afectan al tema y sugerir, en la medida de nuestras posibilidades, algunas respuestas a estas inquietudes, acercando a nuestros lectores experiencias y comentarios que les puedan ser de alguna utilidad.

En los dos números anteriores de esta revista comentamos a grandes rasgos el Logo, lenguaje concebido especialmente con fines didácticos, aunque, por su "potencia" y su alcance, puede perfectamente exceder este marco.

En este número trataremos sobre el "aula de informática". Desearíamos sin embargo que el título de este artículo no induzca a pensar que la informática deba ser tratada necesariamente en un espacio físico determinado, es decir, en un aula; como tampoco desearíamos que los lectores pensaran que el uso de los microordenadores en la escuela deba ser destinado exclusivamente a la enseñanza de la informática.

El uso de ordenadores en la educación crea una problemática nueva y mucho más reciente en España que en otros países. Muy lejos estamos de pensar que sobre el particular esté todo dicho, por lo que, sin el ánimo de creernos los poseedores de la piedra filosofal, expondremos algunos de los criterios que consideramos importantes a la hora de instalar un aula informática en el centro escolar.



La elección de una configuración adecuada

La elección de un determinado equipamiento responde a las necesidades que se deseen satisfacer. En la gestión administrativa de una empresa, según sea el tipo y el volumen del trabajo que requiera ser mecanizado, será la elección del ordenador o de los ordenadores más adecuados, así como los programas y los periféricos necesarios para llevar a cabo las tareas que se determinen. De igual manera, lo que se desee hacer con los ordenadores en la escuela determinará la elección de los mismos.

Descartaremos en principio el uso de los ordenadores para la gestión administrativa de un centro escolar, ya que en este sentido no se diferenciaría mayormente de las necesidades de una empresa. Llevar una contabilidad, las fichas del alumnado, los pagos de cuotas, el envío de cartas, las compras de material, las nóminas, etc., son actividades más o menos comunes a cualquier otra empresa.

Donde sí existen verdaderamente diferencias importantes es en la utilización de los ordenadores para desarrollar una actividad de enseñanza. En este sentido será el "proyecto pedagógico" lo que determinará cuál es el equipamiento más adecuado.

¿Ordenadores o microordenadores?

Los que nos acercamos a la informática por vez primera y nos encontramos poderosamente influidos por el contexto en el que vivimos, tenemos la idea de que un ordenador es un cacharro potente que guarda una enorme cantidad de información, muy parecido a una máquina de escribir y que responde a nuestros requerimientos con una velocidad casi instantánea. También sabemos que algo así es caro y que sólo lo tienen los bancos y empresas o alguna que otra persona adinerada. Por otra parte, tenemos la idea de que un microordenador es un juguete sofisticado en el que podemos librar todo tipo de batallas con seres extraños, al alcance de nuestros bolsillos y que quizá con él podamos empezar a introducirnos en el mundo de la informática.

El "proyecto pedagógico" determinará el equipamiento adecuado

Aunque la mayor parte de esto sea cierto, este conocimiento superficial genera toda una serie de prejuicios que debemos desechar si queremos llevar a cabo nuestra intención de equiparnos para una labor educativa.

No todos los centros escolares pueden equiparse con ciertos ordenadores, que si bien les ofrecen excelentes prestaciones, sus precios son también demasiado elevados para el presupuesto con el que se cuenta. El problema radica en la cantidad de unidades de que se podría disponer para atender a un curso completo (de unos 30 a 35 alumnos). Desalienta la idea de que tengan que haber más de dos alumnos por ordenador, ya que sabemos que en estas condiciones el rendimiento de nuestro trabajo no sería el más adecuado.

Sin embargo, si analizamos más a fondo las prestaciones que debemos exigirle a un ordenador para la enseñanza de la informática en la escuela o para el uso de programas educativos, podremos concluir que una experiencia así está al alcance de prácticamente cualquier escuela.

El primer requisito a exigir es que el ordenador cuente con la posibilidad de soportar lenguajes de programación que puedan ser utilizados por todo el alumnado del centro, desde los más pequeños a los mayores. En resumen: si se trata de un centro de EGB, que no sólo trabaje con BASIC, ya que marginaría a los alumnos de la primera etapa o los limitará a trabajar con unos pocos programas educativos. Un microordenador que funcione también con Logo sería un requisito indispensable. Y aún más: que el Logo, por sus características (ver números de febrero y marzo de TODOSPECTRUM), este en la lengua materna del alumnado.

En cuanto al BASIC es importante que éste sea lo más estándar posible y que su Editor sea simple y rápido. El Spectrum ofrece en este sentido características interesantes, ya que no permite al usuario introducir una línea en el programa si su sintaxis no es correcta. Este control que realiza el ordenador es muy útil con principiantes. ¿A cuántos de nosotros nos ocurre en otros ordenadores que al terminar de escribir un programa y al ejecutarlo comenzamos a leer en la pantalla los errores de sintaxis cometidos? Ya bastantes problemas tenemos en conseguir que los programas hagan lo que deseamos, para tener que luchar también con la sintaxis.

Si bien con estos dos lenguajes se puede cubrir un amplio espectro educativo, sería de desear que el ordenador pudiera utilizar además otros lenguajes de programación; entre ellos el más interesante a nivel educativo es el PASCAL por su valor estructural. Para la Formación Profesional debería determinarse si el COBOL es imprescindible. Desde nuestro punto de vista este es un lenguaje que se está abandonando en la programación de gestión y tampoco pensamos que la labor sea la de formar programadores.



La utilización de lenguajes de programación debe tener un sentido formativo, debe permitir el desarrollo de capacidades intelectuales más que el aprendizaje de la programación.

En cuanto a la "potencia" del ordenador (que de una manera un poco simple medimos en "Kas"), lo más importante es que tenga capacidad suficiente para cargar todos aquellos programas que queramos utilizar con fines educativos.

Un ordenador que cuente en el mercado con un buen catálogo de programas educativos y con una bibliografía extensa con programas comentados que pudiéramos consultar en clase, sería lo más adecuado. También es importante que la mayor parte de estos programas estén en castellano, ya que su máximo aprovechamiento depende de ello.

Si bien, el tiempo de carga de un programa para utilizar en la escuela así como la velocidad de acceso a la información, no es tan crítico como en una empresa, el ordenador debería contar con la posibilidad de utilizar no sólo *cassettes* como fuentes de almacenamiento de información. Aunque el *cassette* sea un medio económico y suficiente para comenzar, debemos pensar que la experiencia tiene que desarrollarse y el ordenador debe soportar este crecimiento. En este sentido, la reciente aparición en el mercado de *diskettes* para el Spectrum pone a este microordenador a la par con otros de si-

SERVICIO DE EJEMPLARES ATRASADOS

Complete su colección de

Todospectrum

A continuación le resumimos el contenido de los ejemplares aparecidos hasta ahora.

Núm. 1 • 250 pts.
Cómo usar el microdrive/Programación Basic/Ampliación Basicare/Rutina despertador/Variables del sistema/Entrada datos mediante máscaras/Protección del software/Sintonice su Spectrum/Programas.

Núm. 2 • 250 pts.
Gráficos profesionales/Desplazamiento pixel a pixel/Utilización de rutinas/Construcción del interface centronics/Programas de utilidad para microdrive/Rutina reset en código máquina/Análisis del editor de textos Tasword/Interfaces para impresoras/Programas.

Núm. 3 • 250 pts.
Novedades sonimag '84/Ampliando el Basic/Programas para ordenar programas/Gráficos con el VU-3D/Lenguaje Forth/Archivos en microdrive/Programación de un interface de impresora/Programas.

Núm. 4 • 250 pts.
De profesión: programador/Consola para el Spectrum/Comparación código máquina-Basic/Análisis programa contabilidad /Calendario/Pascal/Programas.

Núm. 5 • 250 pts.
Floppys para Spectrum/Diseño asistido por ordenador/64 Caracteres por línea/Juego de la vida/Pascal/Así hacemos las portadas/Control de evaluaciones/Programas.

Núm. 6 • 250 pts.
Representación de funciones/Todos los caminos conducen a la ROM/Juegos/Pascal/Construcción de un lápiz óptico/Programas de gestión. El SITI/Logo: tortugas para todos/Interrupciones del Z-80/Programas.





DISPONEMOS DE TAPAS ESPECIALES PARA SUS EJEMPLARES DE

Todospectrum

SIN NECESIDAD DE ENCUADERNACION





(cada tapa es para 6 ejemplares)

Para hacer su pedido, rellene este cupón HOY MISMO y envíelo a:

Todospectrum
Bravo Murillo, 377
Tel. 733 96 62 - 28020 MADRID

Ruego me envíen los siguientes ejemplares atrasados de TODOSPECTRUM .... al precio de 250 pts.

Por favor envíenme .... tapas para la encuadernación de mis ejemplares de TODOSPECTRUM, al precio de 600 pts. más gastos de envío.

El importe lo abonaré
□ POR CHEQUE □ CONTRA REEMBOLSO □ CON MI TARJETA DE CREDITO □ AMERICAN EXPRESS □ VISA □ INTERBANK

Número de mi tarjeta: □□□□□□□□□□□□□□□□□□□□□□□□□□□

Fecha de caducidad .... Firma

NOMBRE ....

DIRECCION ....

CIUDAD .... C. P. ....

PROVINCIA ....

La utilización de lenguajes debe permitir el desarrollo de capacidades intelectuales.

milares características. Por otra parte, el abaratamiento de los cartuchos para el *Microdrive* hace también de este último una instancia intermedia entre el *cassette* y el *diskette* a costes razonables y con prestaciones que mejoran ampliamente las de un *cassette*.

Desde el punto de vista del crecimiento de una experiencia informática en la escuela debemos exigir que nuestra configuración pueda enlazarse en una conexión en red. La primera y más simple utilidad de una red es la posibilidad de carga simultánea de programas en todos los ordenadores. Requisito importante si queremos trabajar con programas educativos o con otro lenguaje diferente al que trae el ordenador de fábrica.

Otra mejora que introduce la red es la posibilidad de un uso compartido de periféricos, como *diskettes* e impresora, con lo que podría reducirse el presupuesto enormemente. Un uso más sofisticado de la misma posibilita la interactividad entre estaciones y podría permitir que el profesor desde un equipo controlara el desarrollo de un determinado trabajo.

Consideramos que la red no es un requisito indispensable en un aula informática, pero la posibilidad de tenerla puede permitirnos mejoras interesantes.

En cuanto a las pantallas podemos exigir que nuestra aula esté equipada o bien con monitores de fósforo o de color o bien con televisores. Un equipo que nos permita elegir entre estas distintas opciones será lo suficientemente flexible como para que determinemos qué es lo más adecuado según nuestro presupuesto.

La calidad gráfica y el uso del color son elementos que ofrecen grandes posibilidades en la enseñanza.

Aunque los monitores están pensados específicamente para un uso informático, no debemos descartar la posibilidad de utilizar televisores, ya que la escuela puede emplearlos con otros fines tales como determinados trabajos en video, etcétera.

Lo que sí es importante es que haya una buena imagen que no dañe la vista y que no desmerezca las cualidades que en este sentido tienen los microordenadores.

Hasta hace poco tiempo se discutía, no sin razón, sobre si un teclado profesional, es decir, similar al de una máquina de escribir, era el que mejores prestaciones ofrecía en la escuela.

No dudamos que la robustez del teclado tiene su importancia ya que los alumnos someten a los equipos a verdaderas "palizas". Pero un teclado de estas características también aumenta el coste de un equipo de una manera importante. Lo que debemos desechar es la idea de que un ordenador en la escuela debe ofrecer las prestaciones de una máquina de escribir. En este aspecto debemos alejarnos de nuestros prejuicios ya que los alumnos jamás tendrán que introducir en el ordenador información en cantidad ni con la rapidez que exige el uso del mismo en una empresa.

Comentaremos ahora un elemento que en general no se suele tener en cuenta y que, sin embargo, según nos relataban los responsables del tema en Gran Bretaña en las Jornadas sobre Informática y Educación, debería apreciarse más. (Jornadas celebradas en noviembre del año paado en Madrid y organizadas por el Ministerio de Educación).

La experiencia pedagógica puede verse francamente beneficiada si el alumnado puede tener en casa el mismo ordenador que también tiene en la escuela. La posibilidad de trabajar con programas que se están utilizando en el colegio, el intercambio de experiencias entre los alumnos y la posibilidad de que ellos generen por su mente programas que puedan utilizarse en la escuela, crea un marco social de aprendizaje de dimensiones interesantes. Este elemento nos decanta hacia equipos con un precio accesible a la familia, siempre y cuando reúnan también el resto de las condiciones.

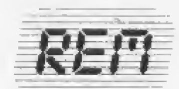
Finalmente, sólo nos queda por comentar la importancia de un mantenimiento barato y rápido.

Recomendaríamos muy especialmente la compra de ordenadores que estén respaldados por una garantía y un servicio técnico fiable. Es cierto que un ordenador puede comprarse sólo por su buen precio; pero cuando se trata de una implantación en la escuela para que todo el alumnado trabaje con ellos no podemos descuidar el servicio postventa. Aquí les recomendamos que se dirijan a profesionales del sector que estén especializados en el tema educativo ya que les podrán asesorar sobre muchas otras cuestiones que implican el uso de un aula informática: la disponibilidad de cierto *software* educativo, el entrenamiento en el uso de los equipos, la formación del profesorado, la confección de programas a medida para realizar algún proyecto determinado, etcétera.

Podemos concluir, por todas estas razones, que un microordenador que cumpla estos requisitos básicos será suficiente para iniciar una experiencia pedagógica con ordenadores en el centro.

Miguel Figini
Miembro del equipo de asesores de Ioshua

Guía del comprador de Todospectrum



• Ordenadores personales Hard y Soft.
• Cursos de Basic.

Oficina **RENOVACION EN MARCHA, S. A.**
C/ Espronceda, 34. 28003-MADRID
Tfno. (91) 441 24 78

REMSHOP 1
Galileo, 4. 28015 MADRID
Tfno. (91) 445 28 08

REMSHOP 2
C/ Dr. Castelo, 14. 28008 MADRID
Tfno. (91) 274 98 43

REMSHOP 3
C/ Modesto Lafuente, 33. 28003 MADRID
Tfno. (91) 233 83 19

REMSHOP BARCELONA
C/ Pelayo, 12. Entresuelo J 08881 BARCELONA
Tfno. (93) 301 47 00

REMSHOP LAS PALMAS
C/ General Mas de Gamindez, 45. LAS PALMAS
Tfno. (928) 23 02 90



ELECTRONICA
SANDOVAL S.A.

DISTRIBUIDORES DE:
COMMODORE-64
ORIC-ATMOS
ZX SPECTRUM
SINCLAIR ZX 81
ROCKWELL'-AIM-65
DRAGON-32
NEW BRAIN
DRAGON-64
CASIO FP-200

ELECTRONICA SANDOVAL, S. A.
C/ SANDOVAL, 3, 4, 6. 28010-MADRID
Teléfonos: 445 75 58 - 445 76 00 - 445 18 70
447 42 01
C/ SANDOVAL, 4 y 6
Centralita 445 18 33 (8 líneas)

CLUB DEL JUEGO

COMPRA — VENTA
PROGRAMAS DE OCASION
ZX 16-48K

Entre otros: Space Raiders, Time Gate, Froggi, Billar Americano, Harrier Attak, Figther Pilot, Tunel 3 D, Styk, Scuba Dive, Base Datos, Ajedrez Cirus y 600 títulos más, pídenos el tuyo.

Por sólo 900 ptas. más gastos de envío, puedes conseguir tu programa preferido, garantizados y comprobados.

Pídenos gratis nuestro catálogo de programas.

Rellena este cupón:
Deseo recibir contra reembolso:
Nombre del programa ....
....
ME LO ENVIAN A:
D. ....
Calle ....
Población ....
Teléfono (si tienes) ....

ENVIAR A: CLUB DEL JUEGO
Apartado Correos 34.155 BARCELONA

COMPUTIQUE

Si posees un Spectrum y —o un QL,
Si dominas el código Máquina,
Si te gusta la programación y
puedes escribir un buen programa

¡CONTACTA CON NOSOTROS!

COMPUTIQUE

C/ Embajadores, 90. 28012 MADRID
Tfno. 227 09 80 - 227 91 99



TODO EN MICROINFORMATICA

SINCLAIR, AMSTRAD
SPECTRAVIDEO, KATSON...
IMPRESORAS, MONITORES, PERIFERICOS
PROGRAMAS EDUCATIVOS, GESTION,
OCIO.

Silva, 5-4.º Tel. 242-24-71
28013 MADRID

CAMAFEO INC.

CASSETTES
DE CALIDAD PROBADA
PARA ORDENADORES

Cada uno	Caja de 10	Caja de 30
C-5 199 ptas	1 393 ptas	3 582 ptas
C-10 209 ptas	1 463 ptas	3 762 ptas
C-15 219 ptas	1 533 ptas	3 942 ptas
C-20 229 ptas	1 602 ptas	4 122 ptas

Libre de gastos de envío contra reembolso correos

CAMAFEO INC. Dep 03

José Lázaro Galdiano, 1. 28036 Madrid.

K-BITS

- APPLE
- COMMODORE
- SPECTRUM
- AMSTRAD
- SPECTRAVIDEO
- DRAGON

OFERTA MES DE ABRIL

En importes superiores a **25.000** ptas. Vale obsequio 10% descuento para su próxima compra

Barquillo, 15 - Tel. 232 57 37
MADRID

FACTURACION SPECTRUM

Un programa que le permite realizar
Facturas
Pedidos
Ofertas
Albaranes
Control de Stocks
Listas de Precios
20 Ficheros diferentes
1000 Artículos
400 Fichas

Un sólo programa de fácil manejo con microdrive con 20 ficheros de clientes, proveedores, artículos, etc.

ALSI, S. A. Antonio López, 154.
Tel. 91/475 43 39. 28026 MADRID



ta-ma

Librería y Editorial Microinformática, le ofrece:

• La mayor variedad de libros de microinformática, capaces de satisfacer todas sus necesidades, ya sean profesionales, familiares, culturales...

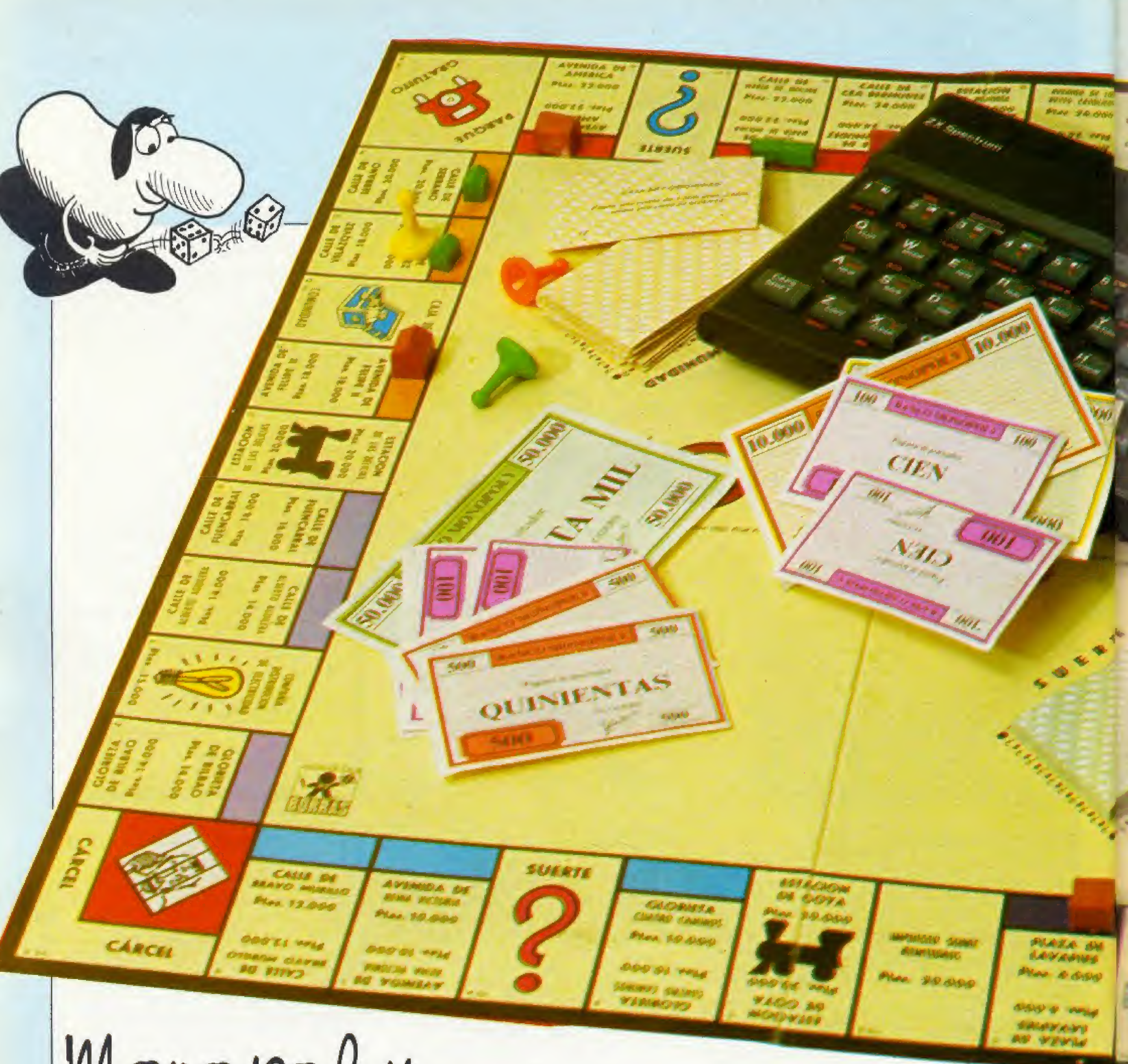
• Todo tipo de información bibliográfica sobre microordenadores, desde AMSTRAD a Sinclair QL; Lenguajes y Técnicas de Programación; Robótica, etc., en cualquier nivel desde el más básico al más avanzado, para sacar el máximo beneficio, diversión y provecho a su ordenador.

Por favor, anoten las materias de su interés.
Ordenadores personales.
Si dispone, especifique marca ....
Aplicación a:
☐ NEGOCIOS. ☐ CASERA. ☐ EDUCACION.
☐ JUEGOS. ☐ OTROS.
Ocupación ....
Comentarios ....
....

NOMBRE ....
DIRECCION ....
CIUDAD ....
CODIGO POSTAL ....
PROVINCIA ....

ta-ma

Libros, Revistas, Suscripciones, Importación y Distribución
Chiquinquirá, 28. Local 4 (Cocuy). 28033 MADRID. Telef.: 764 50 95



Monopoly

Situación: lluviosa tarde de un frío domingo de invierno. Cuatro amigos sin dinero se reúnen mirando sin ver el aburrido programa de nuestra mejor televisión. De repente, uno de ellos sugiere una partida de Monopoly; pero, ilamentable!, no tenemos Monopoly.

Si esta ha sido vuestra situación os sugiero una solución: tomar vuestro Spectrum e introducir el programa que acompaña a este comentario. Para utilizarlo, leer estas instrucciones.

Como sabéis, el Monopoly es un juego de sociedad cuyo objetivo consiste en intentar arruinar a los adversarios mediante diversos negocios.

Una vez hayáis tenido la inmensa paciencia de cargar todo el programa, así como los caracteres gráficos y las matrices que le acompañan, grabaréis el mismo con Save "Monopoly" Line 1. Debéis tener en cuenta que nunca se debe ejecutar con "RUN", pues borraréis las distintas matrices introducidas. Por ello, cuando queráis comenzar una partida debéis utilizar GOTO 5; al hacerlo obtendréis:

"Número de jugadores".

Aquí debéis contestar con un número entre 2 y 4.

Orden de juego

Para establecer el orden de juego, cada jugador irá presionando la tecla "D" haciendo de esta manera funcionar los dados. El jugador que más puntuación saque jugará primero haciéndolo los demás de acuerdo ya con sus posiciones relativas a este jugador. En caso de que se diera un empate por el primer




```
   4 LOAD *"m";1;"dados"CODE USR
 "a",21*8
   5 LET BIO=0: LET cap1=0: LET
NJ=0: LET LIO=0: LET H1=0: LET M
AR=0: LET CAJA=0: DIM f(4): DIM
G(9,4): DIM H(9,6): DIM p(2): DI
M r(38): DIM t(4): DIM z(4): DIM
 S(4): DIM d(5): DIM b$(38,9): D
IM N(2): RANDOMIZE
   7 LET B$(1)="28210": LET B$(2
)="25213": LET B$(3)="22213": LE
T B$(4)="19210": LET B$(5)="1621
9": LET B$(6)="16175": LET B$(7)
="16140": LET B$(8)="16115":, LET
 B$(9)="16085": LET B$(10)="1308
0": LET B$(11)="13118": LET B$(1
2)="13148": LET B$(13)="13178":
LET B$(14)="13219": LET B$(15)="
10217"
   9 LET b$(16)="07210": LET b$(
17)="04217": LET b$(18)="01217":
 LET b$(19)="00170": LET b$(20)=
"00142": LET b$(21)="00110"
  11 LET B$(22)="00082": LET B$(
23)="00052": LET B$(24)="01019":
 LET B$(25)="04016": LET B$(26)=
"07010": LET B$(27)="10016": LET
 B$(28)="13016"
  13 LET B$(29)="16010": LET B$(
30)="19014": LET B$(31)="22014":
 LET B$(32)="25010": LET B$(33)=
"29014"
  15 LET B$(34)="29059": LET B$(
35)="29080": LET B$(36)="29111":
 LET B$(37)="29140": LET B$(38)=
"29171"
  16 FOR I=1 TO 38: LET B$(I,6 T
O 8)="0000": NEXT I
  18 LET D$="
    ": DIM a$(2,8,2)
  20 LET a$(1,1)="AB": LET a$(1,
2)="ER": LET a$(1,3)="KB": LET a
$(1,4)="ET": LET a$(1,5)="KG": L
ET a$(1,6)="MN"
  30 LET a$(2,1)="CD": LET a$(2,
2)="IF": LET a$(2,3)="CL": LET a
$(2,4)="SF": LET a$(2,5)="HL": L
ET a$(2,6)="OP"
  33 POKE 23609,255: GO TO 300
  35 LET x=9: LET y=6: CLS : PRI
NT AT 8,11;"█";AT 9,11;"█"
```

puesto, entra en funcionamiento una rutina de desempate.

Movimiento

Cada jugador irá presionando su número de acuerdo con el orden preestablecido, al hacerlo pone en movimiento los dados, cuando éstos se paren la ficha correspondiente comenzará a moverse al número indicado.

Si la tirada fuera capicúa el jugador volverá a tirar.

Comprar calles

Una vez realizado el movimiento, el jugador tendrá la opción de comprar la calle donde ha caído, siempre y cuando no haya sido previamente adquirida.

Para realizar la operación se presiona la tecla "C", inmediatamente el ordenador realizará la transacción.

Vender calles

Un jugador podrá vender una o varias calles a otro jugador mediante la tecla "V". Al hacerlo, se preguntará qué jugador es el vendedor y cuál el comprador. Después preguntará si lo que se vende es un grupo; si lo es se contestará con el número de la primera calle a vender y el dinero en que fijamos la venta.

Grupos posibles:

(3, 2) (6, 8, 9) (11, 12, 13) (15, 17, 18) (20, 22, 23) (25, 27, 28) (30, 31, 33) (36, 38).

Lo anterior no rige para las estaciones:

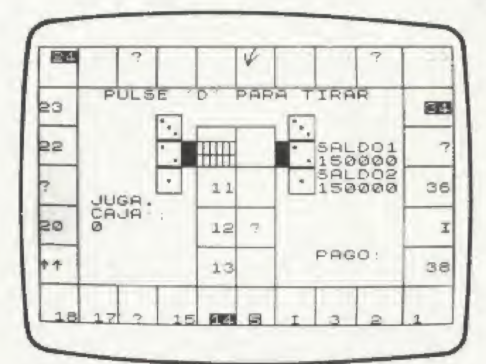
(5, 14, 24, 34).

40

Edificar

Cuando se ha logrado comprar todas las calles que tengan el número del mismo color saldrá un mensaje de grupo completo. A partir de entonces, se podrá edificar. Para ello se pulsará "z", con lo que aparecerá la pregunta:

"Número calle 1 del grupo". A lo que deberemos contestar evidentemente con el número de la primera calle del grupo a edificar.



"Número de casas". Debemos dar un número entre 1 y 4, dependiendo ello del dinero que tengamos.

Una vez colocadas las 4 casas se podrá edificar un hotel y sólo uno por calle del grupo.

Caer en calle de un adversario

Cuando un jugador caiga en una calle de algún adversario deberá pagar el alquiler correspondiente

```
  37 PRINT AT 8,18;"■";AT 9,18;"■"
  40 LET NUM1=3:: LET num=num1: GO SUB 9000
  45 LET x=19: LET num1d=3: GO SUB 9000: LET y=y+1:: GO SUB 9050: LET num=num1d: LET x=9: LET num=num1: GO SUB 9050: LET y=y+1
  50 LET x=9: LET num2=2: LET num=num2: GO SUB 9000
  55 LET x=19: LET num2d=2: LET num=num2d: GO SUB 9000: LET y=y+1: LET num=num2d: GO SUB 9050: LET x=9: LET num=num2: GO SUB 9050: LET y=y+1
  60 LET x=9: LET num3=1: LET NUM=NUM3: GO SUB 9000
  65 LET x=19: LET num3d=1: LET NUM=NUM3d: GO SUB 9000: LET y=y+1: LET num=num3d: GO SUB 9050: LET x=9: LET num=num3: GO SUB 9050
  68 FOR I=1 TO 2
  70 FOR i=1 TO 3: PRINT AT 4+i,4; BRIGHT 1;"   ": PRINT AT 4+i,23; BRIGHT 1;"   ": NEXT i
  93 LET COR=23
  97 NEXT I
 100 PLOT 0,24: DRAW 220,0: PLOT 0,144: DRAW 248,0
 110 PLOT 24,0: DRAW 0,168: PLOT 219,0: DRAW 0,170
 120 PLOT 96,24: DRAW 0,96: DRAW 48,0: DRAW 0,-96: PLOT 120,24: DRAW 0,96
 130 FOR I=1 TO 8
 140 PLOT 24*I,0: DRAW 0,24: PLOT 24*I,145: DRAW 0,25
 150 NEXT I
 160 FOR I=1 TO 5
 170 PLOT 0,I*24: DRAW 24,0: PLOT 219,I*24: DRAW 28,0
 175 NEXT I
 180 FOR I=1 TO 3
 190 PLOT 96,24+I*24: DRAW 48,0
 200 NEXT I
 240 GO SUB 9240
 260 PRINT AT 16,21;"PAGO:";AT 13,4;"CAJA::";AT 12,4;"JUGA."
 290 RETURN
 295 IF NJ<>0 THEN  GO TO 350
 310 INPUT "NUMERO DE JUGADORES: ";NJ
 320 LET p(2)=-1: LET p(1)=1
 335 GO SUB 35
 340 LET JUG=1
 350 LET PY=9: FOR i=1 TO nj
 353 IF LIO=0 THEN  LET D(I)=150000
 355 PRINT AT py-1,21;"SALDO";I
 360 PRINT AT py,21;d(i)
 370 LET py=py+2
 380 NEXT i
 390 PRINT AT 13,4;"CAJA:";AT 14,4;CAJA
 393 IF LIO=1 THEN  RETURN
 400 PRINT  FLASH 1;AT 4,5;"PULSE 'D' PARA TIRAR": LET NJ1=NJ: LET FRU=0
 410 IF INKEY$="D" OR INKEY$="d" AND FRU<=NJ1 THEN  GO SUB 900: LET FRU=FRU+1
 430 IF NJ1=1 THEN  GO TO 460
 440 IF NJ1<>1 AND FRU=NJ1 THEN  LET FRU=0: BEEP 1,7: GO SUB 9300: DIM S(4)
 450 GO TO 410
 460 PRINT AT 4,5;"
           "
 470 FOR i=1 TO nj: LET t(i)=i: NEXT i
 475 LET r(1)=1: LET la=-1: LET ax=4: LET ay=5: PRINT AT 6,5;1;AT 20,28;"■"
 480 FOR i=1 TO nj: PRINT  INVERSE 1;AT ay,ax;i: LET ax=ax+2
 485 IF ax=8 THEN  LET ax=4: LET ay=7
 490 NEXT i
 495 LET BIO=1
 500 IF cap1=1 AND f(hr)<>0 THEN  LET f(hr)=0: LET cap1=0: PRINT AT 17,4;"      ": GO TO 502
 502 IF cap1=1 THEN  LET cap1=2: LET jug=jug-1: PRINT AT 4,5;"Pulse p para tirar": PRINT AT 17,4;"      "
 503 IF d(hr)<0 THEN  GO TO 7000
 504 IF JUG=NJ+1 THEN  LET JUG=1
 505 PRINT AT 12,10; FLASH 1;JUG
 507 IF INKEY$="v" OR INKEY$="V" THEN  GO SUB 7800
 510 IF cap1=0 AND INKEY$=STR$ JUG THEN  BEEP .1,-7: LET 1$=INKEY$: GO SUB 9910
 512 IF cap1=2 AND (INKEY$="p" OR INKEY$="P") THEN  PRINT AT 4,4;o$: LET 1$=STR$ jug: LET capi=0: GO SUB 9910
 515 IF INKEY$="d" OR INKEY$="D" THEN  GO TO 7400
 520 IF INKEY$="i" OR INKEY$="I" THEN  GO SUB 6500: LET LIO=1: GO SUB 35: GO SUB 295: LET LIO=0: GO SUB 8500: GO SUB 2000: GO SUB 8700
 530 IF INKEY$="C" OR INKEY$="c" THEN  GO SUB 6000
 540 IF INKEY$="z" OR INKEY$="Z" THEN  GO SUB 5000
 560 IF INKEY$="H" OR INKEY$="h" THEN  GO TO 7070
 570 IF INKEY$=" " THEN  GO SUB 7500
 600 GO TO 500
 800 STOP
 940 LET fin=50
 950 FOR I=1 TO 2
 960 LET X=10*I-1
 990 LET CON=4: LET FIN=INT (RND*6)+1
1000 FOR j=1 TO fin
1010 LET y=6
1020 LET num=CON: GO SUB 9050: LET NUMO=num
1025 LET Y=Y+1: GO SUB 9000: LET y=y+1
1030 LET NUM=NUM1: GO SUB 9050
1035 LET Y=Y+1: LET NUM=NUM2: GO SUB 9000: LET Y=Y+1
1037 LET NUM=NUM2: GO SUB 9050: LET Y=Y+1
1050 LET num=num3: GO SUB 9000
1060 LET y=6
1070 LET num=NUMO: GO SUB 9000
1080 LET Y=Y+1: LET NUM=NUMO: GO SUB 9050: LET Y=Y+1
1090 LET num=NUM1: GO SUB 9000
1100 LET Y=Y+1: LET NUM=NUM1: GO SUB 9050: LET Y=Y+1
1110 LET num=NUM2: GO SUB 9000
1120 LET Y=Y+1: LET NUM=NUM2: GO SUB 9050
1130 LET NUM3=NUM2: LET NUM2=NUM1: LET NUM1=NUMO
1135 LET CON=CON+1: LET N(I)=NUM2
```

al estado de la calle. Esta operación es realizada automáticamente por el ordenador.

Hipotecar

Cuando un jugador necesite dinero podrá conseguirlo, después de tirar hipotecando una calle de su pertenencia. Para hacer esto existen dos modalidades:

— Se desea hipotecar una calle: sólo hará falta dar el número de ésta.

— Se desea hipotecar una casa de un grupo: se introducirá el número de la primera calle del grupo y a continuación el número de casas a hipotecar.

Recuérdese que el precio de la hipoteca es la mitad que el de compra.

Ningún jugador podrá comprar una calle hipotecada, pero su dueño ya no cobrará alquiler.

Si se desea salir de esta opción se pulsará "0" cuando se pregunte "número de calle a hipotecar".

Existe la opción inversa que se obtiene mediante "D".

Información (I)

Se puede obtener información sobre los títulos de propiedad presionando I.

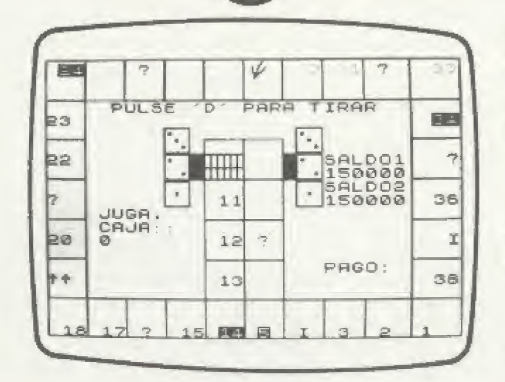
```
1137 IF CON=7 THEN  LET CON=1
1140 NEXT J
1150 NEXT I
1152 LET cap1=0
1155 IF n(1)=n(2) AND Bio<>0 THE
N  LET cap1=1: PRINT AT 17,4; BR
IGHT 1; FLASH 1;"DOBLE": BEEP 1,
25: PAUSE 10: BEEP 1,50
1160 LET TI=N(2)+N(1): PRINT AT
9,5;;"   ";AT 9,5;TI: LET S(FRU+
1)=TI: BEEP .5,7
1170 RETURN
2000 IF lio=2 THEN  GO TO 7350
2002 IF mar=1 THEN  GO TO 5100
2003 LET arc=0: LET al=5: LET ci
fu=4: GO SUB 8100: LET al=5: LET
 cifu=23: GO SUB 8100
2005 IF p(1)=p(2) THEN  LET p(2)
=-1
2010 FOR i=1 TO nj
2020 IF t(i)=p(1) THEN  LET arc=
arc+1: LET z(arc)=i
2030 NEXT i
2040 IF arc<2 THEN  LET p(1)=-1:
 GO TO 2047
2045 IF arc>=2 THEN  LET c1=arc:
 LET cifu=4: LET cif=4: LET al=5
: GO SUB 2120
2047 LET arc=0
2050 FOR i=1 TO nj
2060 IF t(i)=p(2) THEN  LET arc=
arc+1: LET z(arc)=i
2070 NEXT i
2080 IF arc<2 THEN  LET p(2)=-1:
 RETURN
2085 IF arc>=2 THEN  LET c1=arc:
 LET cifu=23: LET cif=23: LET al
=5: GO SUB 2120
2100 RETURN
2120 FOR s=1 TO c1
2130 PRINT  INVERSE 1;AT al,CIF;
z(s)
2140 LET CIF=CIF+2
2150 IF s=2 THEN  LET CIF=CIFU:
LET al=7
2160 NEXT s
2165 LET ar=z(1)
2166 IF t(ar)=0 THEN  PRINT AT 6
,cifu+1;"38": RETURN
2167 PRINT AT 6,cifu+1:t(ar)
2170 RETURN
2250 IF cap1=1 THEN  GO TO 2300
2254 IF f(hr)=0 THEN  GO TO 2300
2255 PRINT AT 4,5;"QUIERE SALIR
DE LA";AT 5,7;" CARCEL?(S/N)"
2260 IF INKEY$="s" OR INKEY$="S"
 THEN  LET pt=5000: GO SUB 8300:
 GO SUB 8310: LET d(hr)=d(hr)-pt
: GO SUB 8320: GO SUB 8340: GO S
UB 8360: LET f(hr)=0: GO TO 2300
2270 IF INKEY$="N" OR INKEY$="n"
 THEN  LET f(hr)=f(hr)-1: PRINT
AT 4,3;O$;AT 5,7;"
": RETURN
2280 GO TO 2255
2300 LET NF=HR
2301 IF t(hr)=38 OR t(hr)=0 THEN
  BEEP 1,7: LET t(hr)=0: LET D(H
R)=D(HR)+20000: PRINT AT 7+2*hr,
21; FLASH 1;d(hr): GO SUB 9950:
PRINT AT 7+2*hr,21;d(hr)
2302 LET ESD=T(HR): LET TI=TI-1:
 LET fin=esd+ti
2303 LET K1=esd
2304 IF fin>=38 THEN  BEEP 1,7:
LET D(HR)=D(HR)+20000: PRINT AT
7+2*hr,21; FLASH 1;d(hr): GO SUB
 9950: PRINT AT 7+2*hr,21;d(hr)
2305 FOR K=ESD TO fin: BEEP .05,
-10
2306 IF K=39 THEN  LET K1=1
2308 IF r(k1+1)=1 THEN : GO TO 8
000
2310 LET t(hr)=t(hr)+1: LET SEA=
T(HR): LET ov=0: GO SUB 9500
2315 IF t(hr)=38 THEN  LET t(hr)
=0: LET k1=-1
2320 LET SEA=T(HR)-1: LET ov=1:
GO SUB 9500
2325 LET k1=k1+1
2330 NEXT K
2332 IF k1=0 THEN  LET r(38)=1:
LET fre=0: GO TO 2340
2335 LET fre=0: LET r(k1)=1
2340 FOR s=1 TO nj
2350 IF t(s)=esd THEN  LET fre=f
re+1: LET z(fre)=s
2360 NEXT s
2365 IF fre=0 AND esd=0 THEN  LE
T r(38)=0: RETURN
2370 IF fre=0 THEN  LET r(esd)=0
: RETURN
2380 IF fre=1 AND esd<>0 THEN  L
ET sea=esd: LET nf=z(1): LET ov=
0: GO SUB 9500: RETURN
2382 IF fre=1 AND esd=0 THEN  LE
T sea=38: LET nf=z(1): LET ov=0:
 GO SUB 9500: RETURN
2385 IF fre>1 AND esd=0 THEN  LE
T sea=38: LET ov=0: LET f$=" ":
GO SUB 9505: RETURN
2390 LET sea=esd: LET f$=" ": LE
T ov=0: GO SUB 9505
2400 RETURN
3000 LET HRP=HR: LET FRA=T(HR)
3002 IF fra=0 THEN  LET fra=38
3003 LET PRO=VAL B$(FRA,6)
3005 LET LON1=K(FRA): LET XA=4+(
20-LON1)/2: PRINT AT 4,XA;C$(FRA
,1 TO LON1)
3006 IF Q(FRA,1)>0 AND Q(FRA,7)=
0 THEN  LET X$=STR$ (Q(FRA,4))+"
 PTS": LET XA=9+(10-LEN X$)/2: P
RINT AT 5,XA;X$: PRINT AT 5,XA;O
$(1 TO 10): GO TO 3010
3007 IF Q(FRA,1)>0 THEN  LET X$=
STR$ (2*Q(FRA,8))+" PTS": LET XA
=9+(10-LEN X$)/2: PRINT AT 5,XA;
X$: BEEP 1,24: PAUSE 25: PRINT A
T 5,9;O$(1 TO 10): GO TO 3010
3008 IF Q(FRA,1)<=0 THEN  PAUSE
100
3015 LET PRO=VAL B$(FRA,6)
3020 IF Q(FRA,1)=-1 OR Q(FRA,1)=
0 THEN  PRINT AT 4,3;O$: PLOT 24
,0: DRAW 0,168: GO TO 3500
3030 IF B$(FRA,6)=STR$ HR OR B$(
FRA,6)="0" OR B$(FRA,9)="H" THEN
 : PRINT AT 4,4;O$: RETURN
3040 IF Q(FRA,6)=0 THEN  LET CA=
G(9,PRO): LET USU=Q(FRA,CA): GO
TO 3065
3062 IF B$(FRA,7)<>"0" THEN  LET
 CA=VAL B$(FRA,7): LET USU=Q(FRA
,1+CA): GO TO 3066
3064 IF B$(FRA,8)<>"0" THEN  LET
 USU=Q(FRA,6): GO TO 3066
3065 IF b$(fra,7)="0" AND b$(fra
,8)="0" THEN  LET USU=Q(FRA,1)
3070 PRINT AT 4,5; FLASH 1;"CALL
E DEL JUGADOR ";PRO
3085 PRINT  BRIGHT 1;AT 17,21;AB
S USU
3087 PRINT AT 7+2*HR,21; FLASH 1
;D(HRP): GO SUB 9950: PRINT AT 7
+2*HR,21;"       "
```

Programas

40

Cárcel

La cárcel es la casilla número 10, sin embargo cuando un jugador va a esta casilla se considera que hace una "visita a la cárcel" y no hay penalización. Cuando un jugador saca una tarjeta de suerte que le mande a la cárcel o cae en la casilla de la flecha, será sancionado con una multa de 5.000 pesetas



o 3 turnos sin tirar. Si en uno de estos 3 turnos al lanzar los dados obtuviera un doble, saldría libre automáticamente.

Bancarrota

Un jugador se puede declarar en bancarrota tecleando "Space".

```
3090 LET D(HRP)=D(HRP)-USU
3100 PRINT AT 7+2*HR,21;"     ";
AT 7+2*hr,21; FLASH 1;D(HRP): GO
 SUB 9950: PRINT AT 7+2*HR,21;D(
HRP)
3120 LET D(PRO)=D(PRO)+USU
3130 PRINT AT 7+2*PRO,21;"
";AT 7+2*pro,21; FLASH 1;D(PRO):
 GO SUB 9950: PRINT AT 7+2*PRO,2
1;D(PRO)
3140 PRINT  FLASH 1;AT 4,5;"OPER
ACION REALIZADA": GO SUB 9950: P
RINT AT 4,3;O$: PLOT 24,0: DRAW
0,168
3150 PRINT AT 17,21;"      "
3200 RETURN
3500 IF T(HR)=10 OR T(HR)=1 THEN
  RETURN
3502 IF t(hr)=29 THEN  GO TO 363
5
3505 IF t(hr)=37 THEN  LET pt1=-
10000: LET caja=caja+10000: GO S
UB 8400: GO TO 8350
3510 IF t(hr)=4 THEN  LET pt1=-2
0000: LET caja=caja+20000: GO SU
B 8400: GO TO 8350
3515 IF T(HR)=19 THEN  LET PT1=C
AJA: LET CAJA=0: GO SUB 8400: GO
 TO 8350
3520 LET pre=INT (27*RND)+1: LET
 long1=v(pre,1): LET long2=v(pre
,2)
3525 LET n$=u$(pre,1,1 TO long1)
: LET m$=u$(pre,2,1 TO long2)
3530 LET x1=3+(24-LEN n$)/2: LET
 x2=3+(24-LEN m$)/2: LET y1=4: L
ET y2=5
3550 PRINT AT y1,x1;n$
3560 PRINT AT y2,x2;m$
3580 GO TO (3580+5*pre)
3585 LET pt1=20000: LET tim=100:
 GO TO 8350
3590 LET tim=100: LET pt1=2500:
GO TO 8350
3595 LET tim=100: LET pt1=10000:
 GO TO 8350
3600 LET hr1=hr: FOR w=1 TO nj
3601 IF w<>hr1 THEN  LET PT=1000
: LET tim=50: LET d(w)=d(w)-pt:
LET hr=w: GO SUB 8300: GO SUB 83
10: GO SUB 8320: GO SUB 8340
3602 NEXT w
3603 LET hr=hr1: LET pt1=nj*1000
: GO TO 8350
3605 LET ti=39-t(hr): GO SUB 225
0: GO SUB 8360: GO TO 2000
3610 LET pt1=5000: LET tim=100:
GO TO 8350
3615 LET pt1=10000: LET tim=100:
 GO TO 8350
3620 LET tim=100: LET pt1=2000:
GO TO 8350
3625 LET ti=40-t(hr): GO SUB 225
0: GO SUB 8360: GO TO 2000
3630 LET pt1=-5000: LET tim=100:
 GO TO 8350
3635 IF t(hr)<10 THEN  LET ti=10
-t(hr): GO SUB 2300: LET f(hr)=3
: GO SUB 8360: GO TO 2000
3636 IF t(hr)>10 THEN  LET ti=48
-t(hr): GO SUB 2300: LET f(hr)=3
: LET pt1=-20000: GO TO 8350
3640 LET pt1=-10000: LET tim=100
: GO TO 8350
3645 LET pt1=10000: LET tim=100:
 GO TO 8350
3650 LET pt1=-5000: LET tim=100:
 GO TO 8350
3655 LET colc=0: LET colh=0: LET
 cas=4000: LET hot=11000
3656 FOR i=1 TO 38
3657 IF VAL b$(i,6)=hr THEN  LET
 colc=colc+VAL b$(i,7): LET colh
=colh+VAL b$(i,8)
3658 NEXT i
3659 LET pt1=-(cas*colc+hot*colh
): LET tim=100: GO TO 8350
3660 LET pt1=-2000: LET tim=100:
 GO TO 8350
3665 LET pt1=10000: LET tim=100:
 GO TO 8350
3670 GO TO 3605
3675 LET pt1=5000: LET tim=100:
GO TO 8350
3680 IF t(hr)<28 THEN  LET ti=28
-t(hr): GO SUB 2250: GO SUB 8360
:: GO SUB 3000: GO TO 2000
3681 IF t(hr)>28 THEN  LET ti=66
-t(hr): GO SUB 2250: GO SUB 8360
: GO SUB 3000: GO TO 2000
3685 LET pt1=-15000: LET tim=100
: GO TO 8350
3690 LET cas=2500: LET hot=3656:
 GO TO 3656
3695 IF t(hr)<27 THEN  LET ti=27
-t(hr): GO SUB 2250: GO SUB 8360
: GO SUB 3000: GO TO 2000
3696 IF t(hr)>28 THEN  LET ti=65
-t(hr): GO SUB 2250: GO SUB 8360
: GO SUB 3000: GO TO 2000
3700 IF t(hr)<11 THEN  LET ti=11
-t(hr): GO SUB 2250: GO SUB 8360
: GO SUB 3000: GO TO 2000
3701 IF t(hr)>11 THEN  LET ti=49
-t(hr): GO SUB 2250: GO SUB 8360
: GO SUB 3000: GO TO 2000
3705 IF t(hr)<8 THEN  LET ti=8-t
(hr): GO SUB 2250: GO SUB 8360:
GO SUB 3000: GO TO 2000
3706 IF t(hr)>8 THEN  LET ti=46-
t(hr): GO SUB 2250: GO SUB 8360:
 GO SUB 3000: GO TO 2000
3710 LET pt1=15000: LET tim=100:
 GO TO 8350
3715 GO TO 3635
5000 LET H1R=HR: LET H1=0
5010 INPUT "NUMERO CALLE 1 DEL G
RUPO ";NP
5020 IF B$(NP,7)="4" THEN  LET H
1=1: LET NC=1: GO TO 5070
5050 INPUT "NUMERO DE CASAS ";NC
5070 LET i1=0: LET can=3: LET IN
I=VAL B$(NP,5)
5072 IF INI=0 THEN  PRINT AT 4,4
;"OPERACION NO PERMITIDA": PAUSE
 100: PRINT AT 4,4;O$: RETURN
5073 IF H(INI,3)=1 THEN  PRINT A
T 4,3;"OPERACION NO PERMITIDA":
PAUSE 100: PRINT AT 4,3;O$: RETU
RN
5075 IF G(INI,HR)<>3 AND (INI<>1
 AND INI<>3) THEN  PRINT AT 4,4;
"OPERACION NO PERMITIDA": PAUSE
100: PRINT AT 4,4;O$: RETURN
5080 IF inI=3 OR ini=1 THEN  LET
 can=2
5085 IF G(INI,HR)<>2 AND (INI=1
OR INI=3) THEN  PRINT AT 4,4;"OP
ERACION NO PERMITIDA": PAUSE 100
: PRINT AT 4,4;O$: RETURN
5086 FOR I=1 TO CAN
5087 IF Q(NP+I1,1)<=0 THEN  LET
I1=I1+1: LET I=I-1: GO TO 5080
5088 IF B$(NP+I1,9)="H" THEN  PR
INT AT 4,4;"Operacion no permiti
da": BEEP 1,23: PAUSE 25: PRINT
AT 4,4;o$: RETURN
5089 NEXT i
5090 LET PT1=-NC*Q(NP,7)*can
```

OFERTA DE SUSCRIPCION

Todospectrum

REVISTA EXCLUSIVA PARA USUARIOS

Te ofrece la posibilidad de suscribirte con unas condiciones muy ventajosas para ti:

1 *Recibir puntualmente, en tu domicilio la publicación TODOSPECTRUM que durante 12 meses te proporcionará lectura, programas, ayuda, entretenimiento, etc.*

2 *Consigue un práctico regalo:*



Una obra fundamental en la biblioteca de los aficionados al Spectrum:

- *Reglas y herramientas del Basic.*
- *La técnica de los organigramas.*
- *El fabuloso mundo de las rutinas.*
- *Variables y cadenas.*
- *Funciones matemáticas usuales.*

Esta publicación, escrita con estilo ameno y práctico, te ayudará a sacar todo el partido posible a tu máquina.

3 *La opción de ser protagonista. Tú puedes tener una participación directa con tus comentarios, programas, sugerencias, etc.*

4 *Obtener premios importantes con tus programas, y temas de interés.*

EN DEFINITIVA, TODO SON VENTAJAS

No dejes pasar esta oportunidad, suscríbete a "TODOSPECTRUM", cumplimentando hoy mismo el cupón de respuesta adjunto.

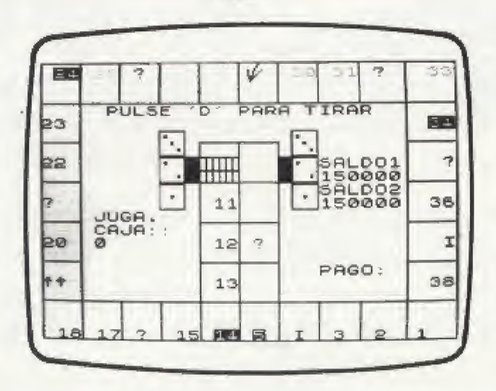


BRAVO MURILLO, 377 - 5.° A
TELEFONO: 733 74 13/47/63/97 28020 MADRID

Programas



Al hacer esto aparece un mensaje de seguridad, si éste se contesta afirmativamente se procede a retirar el saldo de este jugador pasando a ocupar su número el jugador que tuviera el número superior a él. Por ejemplo, si están jugando 4 jugadores y se retira el 2 ocurriría que el 1 seguiría siendo el 1, pero el pasará a ser el 2 y el 4 pasará a ser el 3.

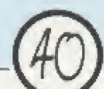


Matrices

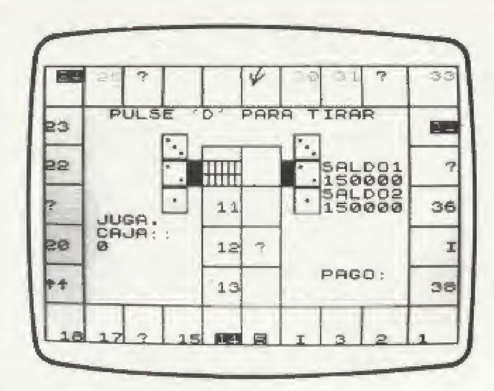
Las matrices listadas (ver datos anexos) han de cargarse independientemente del listado. Estas matrices serán luego grabadas permaneciendo de este modo en cinta, pese a que no estén en el listado. Observe que en el listado se carga esta información directamente de microdrive (línea 4).

```
5095 IF d(hr)+pt1<0 THEN  PRINT
AT 4,8; BRIGHT 1;"NO HAY FONDOS"
: BEEP 1,17: PAUSE 50: PRINT AT
4,8;O$: RETURN
5097 LET MAR=1: GO TO 8350
5100 LET I1=0: LET mar=0: FOR i=
1 TO can
5110 IF q(np+i1,1)<=0 THEN  LET
i1=i1+1: LET i=i-1: GO TO 5130
5115 IF h1=1 THEN  LET b$(np+i1,
7)="0": LET b$(np+i1,8)="1": LET
 i1=i1+1: GO TO 5130
5120 LET cras=VAL b$(np+i1,7)+nc
: LET b$(np+i1,7)=STR$ cras
5125 LET i1=i1+1
5130 NEXT i
5200 IF NP<=5 OR (np>=14 AND np<
=18) OR (NP>=24 AND NP<33) THEN
 LET OX=VAL B$(NP,1 TO 2)-1: LET
 OY=VAL B$(NP,3 TO 4)
5210 IF (NP>18 AND NP<24) OR (NP
>=33 AND NP<=38) THEN  LET OX=VA
L B$(NP,1 TO 2): LET OY=VAL B$(N
P,3 TO 4)+1
5220 IF np>5 AND np<10 THEN  LET
 OX=VAL B$(NP,1 TO 2)-1: LET OY=
VAL B$(NP,3 TO 4)+1
5230 IF np>10 AND np<14 THEN  LE
T OX=VAL B$(NP,1 TO 2)+1: LET OY
=VAL B$(NP,3 TO 4)+1
5245 IF (np>10 AND np<14) OR NP<
=5 OR (np>=14 AND np<=18) OR (NP
>33 AND NP<=38) THEN  LET VAR=-1
5246 IF (NP>18 AND NP<=33) OR (n
p>5 AND np<10) THEN  LET VAR=1
5247 IF H1=0 THEN  LET CAR=VAL B
$(NP,7)
5248 IF H1=1 THEN  LET CAR=4
5250 LET H(INI,1)=OX: LET H(INI,
2)=OY: LET H(INi,3)=H1: LET H(IN
I,6)=NP: LET OX1=OX: LET OY1=OY
5251 LET H(INI,4)=CAR: LET H(INI
,5)=VAR: GO TO 5253
5252 LET OX1=H(INI,1): LET OY1=H
(INI,2): LET H1=H(INI,3): LET CA
R=H(INI,4): LET VAR=H(INI,5): LE
T NP=H(INI,6)
5253 FOR J=1 TO G(INI,H1R)
5255 LET crom=0
5260 FOR I=1 TO CAR
5270 IF np<=5 OR (np>=14 AND np<
=18) OR (np>=24 AND np<=33) THEN
  PLOT OX1*8,168-Oy1*8-var*crom:
 DRAW 0,4: DRAW 4,0: DRAW 0,-4:
DRAW -4,0: IF H1=1 THEN  PRINT A
T Oy1,Ox1;"▙"
5280 IF (np>5 AND np<14) OR (np>
18 AND np<24) OR (np>33 AND np<=
38) THEN  PLOT OX1*8.25+var*crom
,168-Oy1*8: DRAW 0,4: DRAW 4,0:
DRAW 0,-4: DRAW -4,0: IF H1=1 TH
EN  PRINT AT Oy1,Ox1;"▙"
5285 LET crom=crom+5
5290 NEXT I
5300 IF NP<=5 OR (np>=14 AND np<
=18) OR (NP>=24 AND NP<=33) THEN
  LET OX1=OX1+3*VAR: IF q(np+j,1
)<=0 THEN  LET ox1=ox1+3*var
5310 IF (np>5 AND np<14) OR (NP>
18 AND NP<24) OR (NP>33 AND NP<=
38) THEN  LET OY1=OY1-VAR*3: IF
q(np+j,1)<=0 THEN  LET oy1=oy1-v
ar*3
5320 NEXT j
5500 BEEP 1,14: RETURN
6000 LET FRA=t(hr)
6010 IF FRA=0 THEN  LET FRA=38
6060 PRINT AT 4,7; FLASH 1;"ORDE
N DE COMPRA": BEEP 1,7
6070 IF Q(FRA,1)=-1 OR Q(FRA,1)=
0 THEN  PRINT  FLASH 1;AT 4,4;"O
PERACION NO PERMITIDA": BEEP 1,6
: PRINT AT 4,4;O$: RETURN
6075 IF B$(FRA,6)<>"0" THEN  PRI
NT  FLASH 1;AT 4,5;"CALLE DEL JU
GADOR ";B$(FRA,6): BEEP 1,6: PRI
NT AT 4,4;O$: RETURN
6080 GO SUB 8200
6085 IF D(HR)-PRI<0 THEN  PRINT
AT 4,4;O$;AT 4,8;"NO HAY FONDOS"
: BEEP 1,7: PRINT AT 4,4;O$: RET
URN
6090 LET B$(FRA,6)=STR$ HR
6095 LET pt=pri: LET tim=100: GO
 SUB 8300
6100 GO SUB 8310
6110 LET D(HR)=D(HR)-PRI
6120 GO SUB 8320
6140 PRINT  FLASH 1;AT 4,5;"OPER
ACION REALIZADA": BEEP 1,5: PRIN
T AT 4,4;O$
6150 LET IN1=VAL B$(FRA,5)
6160 LET G(IN1,HR)=G(IN1,HR)+1
6165 IF G(IN1,HR)=2 AND IN1<>1 A
ND in1=3 THEN  PRINT  BRIGHT 1;A
T 4,7;"GRUPO COMPLETO": BEEP 1,7
: BEEP 1,7: PRINT AT 4,4;O$
6170 IF G(IN1,HR)=3 AND IN1<>9 A
ND IN1<>3 AND IN1<>1 THEN  PRINT
  BRIGHT 1;AT 4,7;"GRUPO COMPLET
O": BEEP 1,7: BEEP 1,7: PRINT AT
 4,4;O$
6180 IF G(IN1,HR)=4 AND IN1=9 TH
EN  PRINT  BRIGHT 1;AT 4,7;"GRUP
O COMPLETO": BEEP 1,7: BEEP 1,7:
 PRINT AT 4,4;O$
6190 PRINT  OVER 1;AT 17,21;PRI
6200 RETURN
6550 CLS : LET JUGA=1
6560 PRINT AT 8,3;"1 - VER TODOS
 LOS TITULOS"
6570 PRINT AT 11,3;"2 - VER UN
TITULO"
6580 PRINT AT 14,3;"3 - VER TITU
LOS DE UN JUGADOR"
6585 PRINT AT 17,3;"4 - SALIR"
6590 INPUT "OPCION: ";OP
6595 IF OP=4 THEN  RETURN
6600 LET PRIN=1: LET FIN1=38
6610 IF OP=2 THEN  INPUT "CUAL?
";CALL: LET PRIN=CALL: LET FIN1=
CALL
6620 IF OP=3 THEN  INPUT "CUAL?
";JUGA
6690 LET caos=1
6700 FOR I=PRIN TO FIN1
6705 IF OP=3 AND VAL B$(I,6)<>JU
GA THEN  GO TO 6800
6710 CLS : LET LOND=K(I)
6715 LET IN=VAL B$(I,5): LET INV
=0: LET BR=0
6720 LET XA=4+(20-LOND)/2
6725 IF IN=9 THEN  LET INV=1
6730 IF IN=8 THEN  LET IN=3: LET
 BR=1
6735 IF IN=7 THEN  LET IN=5: LET
 BR=1
6740 PRINT  INK IN; BRIGHT BR; I
NVERSE INV;AT 3,XA-3;C$(I,1 TO L
OND);"  ";I: PRINT : PRINT
6742 IF Q(I,1)<=0 THEN  GO TO 67
87
6743 IF IN=9 THEN  GO SUB 6850:
GO TO 6785
6745 PRINT "Precio de compra: ";
TAB 20;2*q(i,8): PRINT D$(1)
6747 PRINT "Por pasar: ";TAB 20;
q(i,1)
6750 FOR J=2 TO 6
```

```
6760 PRINT D$(J);TAB 20;caos*q(i
,J)
6770 NEXT J
6772 PRINT D$(7)
6774 PRINT D$(8);TAB 20;Q(I,7):
PRINT D$(9);Q(I,7)
6776 PRINT D$(10): PRINT D$(11);
TAB 20;Q(I,8)
6785 PRINT : PRINT : PRINT "PROP
IEDAD DEL JUGARDOR  "; B$(I,6)
6786 IF op=3 THEN  PRINT #1;"Pul
se una telca": PAUSE 0: GO TO 68
00
6787 IF op=1 THEN  PRINT #1;"Pul
se una tecla para continuar R pa
ra retroceder S para salir": PAU
SE 0
6790 IF INKEY$="S" OR INKEY$="s"
 THEN  GO TO 6500
6795 IF INKEY$="R" OR INKEY$="r"
 AND i>1 THEN  LET i=i-2: GO TO
6800
6797 IF INKEY$="R" OR INKEY$="r"
 AND i=1 THEN  LET i=37
6800 NEXT I
6810 IF OP=2 THEN  PRINT #1;"Pul
se una tecla":  PAUSE 0
6840 GO TO 6500
6850 PRINT "Precio de compra";TA
B 20;20000: PRINT "ALQUILER"
6860 PRINT "Con 1 estacion";TAB
20;2500: PRINT "Con 2 estaciones
";TAB 20;5000
6870 PRINT "Con 3 estaciones";TA
B 20;10000: PRINT "Con 4 estacio
nes";TAB 20;20000: PRINT "HIPOTE
CA:";TAB 20;10000
6880 RETURN
6900 STOP
7060 PRINT AT 4,7;"Sin fondos, d
ebe hipotecar.": LET litu=1
7070 INPUT "Que calle desea hipo
tecar ";nuc
7073 IF nuc=0 THEN  GO TO 500
7074 IF nuc<=0 THEN  LET jab=hr:
 GO TO 7520
7075 IF VAL b$(nuc,6)<>hr THEN
PRINT AT 4,4;"esa calle no te pe
rtenece": BEEP 1,25: PAUSE 25: P
RINT AT 4,4;o$: GO TO 7070
7077 IF b$(nuc,9)="H" THEN  PRIN
T AT 4,6;"Calle ya hipotecada":
BEEP 1,12: PRINT AT 4,4;o$: GO T
O 7070
7080 IF b$(nuc,7)<>"0" OR b$(nuc
,8)<>"0" THEN  GO TO 7200
7100 IF q(nuc,6)=0 THEN  LET pt1
=10000: LET b$(nuc,9)="H": PRINT
 AT VAL b$(nuc,2 TO 4),VAL b$(nu
c,1 TO 2);"H": GO TO 8350
7110 LET pt1=q(nuc,8): LET b$(nu
c,9)="H": PRINT AT VAL b$(nuc,3
TO 4),VAL b$(nuc,1 TO 2);"H": GO
 SUB 9289: GO TO 8350
7200 IF b$(nuc,8)<>"0" THEN  LET
 max=5: GO TO 7230
7210 LET max=VAL b$(nuc,7)
7230 PRINT AT 4,5;"Maximo numero
 de casas";AT 5,7;"a hipotecar";
max
7240 INPUT "Cuantas casas hipote
cas: ";ven
7245 LET ini=VAL b$(nuc,5): LET
gru=3
7250 IF b$(nuc,5)="1" OR b$(nuc,
5)="3" THEN  LET gru=2
7255 LET ia1=0
7260 FOR i=1 TO gru
7270 IF q(nuc+ia1,1)<=0 THEN  LE
T ia1=ia1+2: GO TO 7300
7280 IF max<5 THEN  LET b$(nuc+i
a1,7)=STR$ (max-ven): LET h(ini,
4)=max-ven
7290 IF max=5 THEN  LET b$(nuc+i
a1,7)=STR$ (5-ven): LET b$(nuc+i
a1,8)="0": LET h(ini,3)=0: LET h
(ini,4)=5-ven
7295 LET ia1=ia1+1
7300 NEXT i
7310 LET pt1=q(nuc,7)*ven/2*gru
7320 LET lio=2: GO TO 8350
7350 LET lio=1: GO SUB 35: GO SU
B 295: LET lio=0: GO SUB 8500: G
O SUB 2000: GO SUB 8700
7360 GO TO 7070
7400 INPUT " CALLE A DESHIPOTECA
R ";cal
7403 IF CAL=0 THEN  RETURN
7405 IF q(cal,1)=0 OR b$(cal,9)<
>"H" THEN  PRINT AT 5,4;"OPERACI
ON NO PERMITIDA": BEEP 1,15: PAU
SE 25: GO TO 7400
7406 LET IN=VAL B$(CAL,5)
7407 IF VAL b$(nuc,6)<>hr THEN
PRINT AT 4,4;"ESA CALLE NO ES TU
YA": BEEP 1,25: PAUSE 25: PRINT
AT 4,4;o$: GO TO 7400
7408 LET IN=VAL B$(I,5): LET BR=
0: LET INV=0: LET XP=VAL B$(I,1
TO 2): LET YP=VAL B$(I,3 TO 4)
7409 IF IN=9 THEN  LET INV=1
7410 IF IN=7 THEN  LET IN=5: LET
 BR=1
7415 IF q(nuc,6)=0 THEN  LET pt1
=-10000: LET b$(CAL,9)="0": PRIN
T  INK IN; BRIGHT BR; INVERSE IN
V;AT VAL b$(CAL,2 TO 4),VAL b$(C
AL,1 TO 2);CAL: GO SUB 9289: GO
TO 8350
7420 LET PT1=-Q(CAL,8): LET B$(C
AL,9)="0": PRINT  INK IN; BRIGHT
 BR; INVERSE INV;AT VAL B$(CAL,3
 TO 4),VAL B$(CAL,1 TO 2);CAL: G
O SUB 9289: GO TO 8350
7500 PRINT AT 4,10;"SEGURO (S/N)
"
7510 IF INKEY$="S" OR INKEY$="s"
 THEN  PRINT AT 4,7;"Espere por
favor": GO TO 7520
7512 IF INKEY$="N" OR INKEY$="N"
 THEN  PRINT AT 4,4;o$: RETURN
7513 GO TO 7510
7515 IF jug=nj THEN  LET jab=1
7517 LET jab=jug+1
7520 FOR I=1 TO 38
7530 IF B$(I,6)<>STR$ JAB THEN
GO TO 7560
7540 LET INI=VAL B$(I,5): LET B$
(I,6 TO 9)="000"
7550 FOR W=1 TO 4: LET H(INI,W)=
0
7560 LET G(INI,JAB)=0
7570 NEXT I
7571 GO SUB 7750
7573 FOR I=JAB TO NJ-1: LET T(I)
=T(I+1): LET D(I)=D(I+1): NEXT I
7575 IF JAB=NJ THEN  LET JUG=1:
GO TO 7700
7580 FOR I=1 TO 38
7590 LET B6=VAL B$(I,6)
7600 IF B6>=JAB THEN  LET B$(I,6
)=STR$ (B6-1)
7610 NEXT I
7620 FOR I=1 TO 9
7630 FOR J=JAB TO NJ-1
7640 LET G(I,J)=G(I,J+1)
7650 NEXT J
7700 LET NJ=NJ-1
7710 LET LIO=1: GO SUB 35: GO SU
B 295: LET LIO=0: GO SUB 8500: G
O SUB 2000: GO SUB 8700
7720 RETURN
7750 LET com=0
7755 FOR i=1 TO nj
7760 IF i=hr THEN  GO TO 7770
7765 IF t(i)=t(hr) THEN  LET com
=com+1
7770 NEXT i
7775 IF com=1 AND p(1)=t(hr) THE
N  LET p(1)=0
7780 IF com=1 AND p(2)=t(hr) THE
N  LET p(2)=0
7785 RETURN
7800 INPUT "De jugador ";jugv
7810 INPUT "a jugador ";jugc
7820 INPUT "Es un grupo? ";j$
7830 IF j$<>"n" AND j$<>"N" THEN
  GO TO 7900
7840 INPUT "Numero de calle ";nu
mc
7850 INPUT "Valor de la transacc
ion ";tran
7860 LET ini=VAL b$(numc,5)
7870 LET g(ini,jugv)=g(ini,jugv)
-1
7880 LET g(ini,jugc)=g(ini,jugc)
+1
7890 IF (ini=1 OR ini=3) AND g(i
ni,jugc)=2 THEN  PRINT  BRIGHT 1
;AT 4,7;"Grupo completo": BEEP 1
,18
7893 IF ini<>1 AND ini<>3 AND g(
ini,jugc)=3 THEN  PRINT  BRIGHT
1;AT 4,7;"Grupo completo": BEEP
1,18
7895 LET b$(numc,6)=STR$ jugc: L
ET usu=-tran: LET PRO=JUGC: LET
HRP=JUGV: GO SUB 3080: RETURN
7900 INPUT "Valor de la transacc
ion ";tran
7905 LET vr=VAL j$: LET ini=VAL
b$(vr,5): LET g(ini,jugv)=0
7910 IF ini=1 OR ini=3 THEN  LET
 g(ini,jugc)=2: LET on=2
7920 IF ini<>1 OR ini<>3 THEN  L
ET g(ini,jugc)=3: LET on=3
7925 LET ia1=0
7930 FOR i=1 TO on
7935 IF q(vr,1)=0 THEN  LET ia1=
ia1+1: LET i=i-1: GO TO 7930
7940 LET b$(vr,6)=STR$ jab
7950 NEXT i
7960 LET usu=-tran: LET pro=jugc
: LET hrp=jugv: GO SUB 3080: RET
URN
8000 LET fw=1: LET k1r=k1
8010 FOR i=2 TO nj
8011 LET esla=k1r+i
8013 IF esla>=39 THEN  LET esla=
```




```
esla-38
8014 IF r(esla)=0 THEN  GO TO 80
23
8015 IF r(esla)=1 THEN  LET fw=f
w+1
8020>NEXT i
8023 IF k+fw>=fin+1 THEN  GO TO
9600
8025 LET kkf=esla
8050 IF t(hr)+fw<37 THEN  LET ov
=1: LET sea=t(hr): GO SUB 9500:
LET t(hr)=t(hr)+fw+1: LET ov=0:
LET sea=t(hr): GO SUB 9500: LET
k=k+fw: LET k1=k1+fw+1: GO TO 23
30
8060 IF t(hr)+fw>=38 THEN  LET o
v=1: LET sea=t(hr): GO SUB 9500:
 LET t(hr)=kkf: LET ov=0: LET se
a=t(hr): GO SUB 9500: LET k1=kkf
: LET k=k+fw: GO TO 2330
8070 IF t(hr)+fw=37 THEN  LET se
a=t(hr): LET ov=1: GO SUB 9500:
LET t(hr)=38: LET ov=0: LET sea=
38: GO SUB 9500: LET k1=0: LET t
(hr)=0: LET k=k+fw: GO TO 2330
8100 FOR i=0 TO 2: PRINT  BRIGHT
 1;AT al+i,cifu;"    ": NEXT i: R
ETURN
8200 IF Q(FRA,6)=0 THEN  LET PRI
=2*Q(FRA,5): RETURN
8210 LET PRI=Q(FRA,8)*2
8220 RETURN
8300 PRINT  BRIGHT 1;AT 17,21;AB
S Pt: RETURN
8310 PRINT AT 7+2*HR,21; FLASH 1
;D(HR): GO SUB 9950: PRINT AT 7+
2*HR,21;"        ": RETURN
8320 PRINT AT 7+2*HR,21; FLASH 1
;D(HR): GO SUB 9950
8330 PRINT AT 7+2*HR,21;D(HR): R
ETURN
8340 PRINT  OVER 1;AT 17,21;ABS
Pt: RETURN
8350 LET pt=pt1: GO SUB 8300: GO
 SUB 8310: LET d(hr)=d(hr)+pt: G
O SUB 8320: GO SUB 8360: GO SUB
8340: GO TO 1199
8360 PRINT AT y1,3;o$+" ": PRINT
 AT y2,3;o$+" ": PLOT 24,0: DRAW
 0,168
8370 PRINT  FLASH 1;AT 4,5;"OPER
ACION REALIZADA": PAUSE 50: PRIN
T AT 4,4;O$: RETURN
8400 PRINT  FLASH 1;AT 14,4;CAJA
: LET TIM=50: GO SUB 9950: PRINT
 AT 14,4;CAJA: RETURN
8507 FOR T=1 TO NJ
8510 LET SEA=T(T)
8520 LET NF=T: GO SUB 9500
8530 NEXT T
8540 IF P(1)>0 THEN  LET ov=0: L
ET SEA=P(1): LET F$=" ": GO SUB
9505
8550 IF P(2)>0 THEN  LET ov=0: L
ET SEA=P(2): LET F$=" ": GO SUB
9505
8560 RETURN
8605 LET ti=39-t(hr)
8700 FOR V=1 TO NJ
8702 LET H1R=V
8705 FOR W=1 TO 9
8710 IF H(W,4)=0 THEN  GO TO 873
0
```




```
8720 LET INI=W: GO SUB 5252
8730 NEXT W
8735 NEXT V
8740 RETURN
9000 PRINT AT y,x;a$(1,num): RET
URN
9050 PRINT AT Y,X;A$(2,NUM): RET
URN
9240 FOR I=1 TO 38
9245 LET IN=VAL B$(I,5): LET BR=
0: LET INV=0: LET XP=VAL B$(I,1
TO 2): LET YP=VAL B$(I,3 TO 4)
9250 IF IN=9 THEN  LET INV=1
9265 IF IN=7 THEN  LET IN=5: LET
 BR=1
9267 IF b$(i,9)="H" THEN  PRINT
 INK IN; INVERSE INV; BRIGHT BR;
AT YP,XP;"H": GO TO 9280
9270 PRINT  INK IN; INVERSE INV;
 BRIGHT BR;AT YP,XP;I
9280 NEXT I
9282 PRINT AT VAL b$(7,3 TO 4),V
AL b$(7,1 TO 2);"? ";AT 1,16;"
";AT 21,19;"I";AT 14,29;" I";AT
8,13;"  ";AT 21,7;"? ";AT 11,0;"
? ";AT 1,7;"? ";AT 1,25;"? ";AT
8,29;" ?"
9283 PRINT AT 17,0;"#"
9284 PLOT 96,103: DRAW 24,0: PLO
T 96,111: DRAW 24,0
9286 FOR I=96 TO 120 STEP 4: PLO
T I,96: DRAW 0,15: NEXT I
9288 PLOT 132,170: DRAW -5,-12:
DRAW -1,7: DRAW 1,-7: DRAW 7,7
9289 PLOT 0,0: DRAW 249,0: DRAW
0,170: DRAW -249,0: DRAW 0,-170
9298 RETURN
9300 LET W1=1: LET NJ2=NJ1: LET
COR1=1: LET MIN=S(1): LET MAX=S(
1)
9310 FOR W=1 TO NJ2
9320 IF S(W)>MAX THEN  LET MAX=S
(W): LET W1=W
9330 IF S(W)<MIN THEN  LET MIN=S
(W)
9340 NEXT W
9350 FOR W=W1+1 TO NJ2
9360 LET S(W-1)=S(W)
9370 NEXT W
9380 IF NJ2=2 THEN  LET s(2)=max
: LET s(1)=min: GO TO 9410
9390 LET S(NJ2)=MAX: LET MAX=MIN
: LET NJ2=NJ2-1
9400 GO TO 9310
9410 FOR W=NJ1-1 TO 1 STEP -1
9420 IF S(NJ1)<>S(W) THEN  LET N
J1=COR1: PRINT AT 4,4;"
        ";AT 4,6;"DESEMPATE
ENTRE ";NJ1: RETURN
9430 LET COR1=COR1+1
9440 NEXT W
9450 LET nj1=cor1: PRINT AT 4,4;
"                    ";AT 4,6
;"DESEMPATE ENTRE ";NJ1: RETURN
9500 LET f$=STR$ nf
9505 IF SEA>=39 THEN  LET SEA=39
-38
9507 LET pos=SEA
9510 IF SEA=-1 THEN  LET POS=37:
 LET OV=1: LET SEA=37
9520 IF SEA=0 THEN  LET OV=1: LE
T POS=38: LET SEA=38
9530 IF SEA<=18 THEN  PRINT  OVE
R ov;AT VAL b$(pos,3 TO 4)-1,VAL
 b$(pos,1 TO 2); INVERSE 1;f$
9540 IF SEA>18 AND sea<24 THEN
PRINT  OVER ov;AT VAL b$(pos,3 T
O 4),VAL b$(pos,1 TO 2)+2; INVER
SE 1;f$
9550 IF SEA>23 AND sea<34 THEN
PRINT  OVER ov;AT VAL b$(pos,3 T
O 4)+1,VAL b$(Pos,1 TO 2); INVER
SE 1;f$
9560 IF SEA>33 THEN  PRINT  OVER
 ov;AT VAL b$(pos,3 TO 4),VAL b$
(pos,1 TO 2)-1; INVERSE 1;f$
9570 RETURN
9600 IF fin+1=38 THEN  LET k1=38
: LET fin=-1: LET t(hr)=0: LET s
ea=k: LET ov=1: GO SUB 9500: LET
 sea=38: LET ov=0: LET f$=" ": G
O SUB 9505: GO TO 9620
9602 IF fin>=38 THEN  LET fin=fi
n-38
9605 LET k1=k1+1: LET t(hr)=fin+
1
9610 LET sea=k1-1: LET ov=1: GO
SUB 9500: LET sea=fin+1: LET ov=
0: LET f$=" ": GO SUB 9505
9620 IF p(1)=-1 THEN  LET p(1)=f
in+1: GO TO 2335
9630 IF p(2)=-1 THEN  LET p(2)=f
in+1
9670 GO TO 2335
9750 LET c1=2: LET z(1)=hr
9755 FOR hr=s TO nj
9760 IF hr=s THEN  GO TO 9780
9770 IF t(hr)=t(s) THEN  LET z(c
1)=s: LET c1=c1+1
9780 NEXT s
9790 LET c1=c1-1: RETURN
9800 IF p(1)=t(hr) THEN  LET p(1
)=0
9810 IF p(2)=t(hr) THEN  LET p(2
)=0
9820 RETURN
9900 PRINT AT 7+2*HR,21;"      ";A
T 7+2*HR,21;D(HR): RETURN
9910 IF L$=STR$ JUG THEN  PRINT
AT 4,4;O$: LET JUG=JUG+1: LET hr
=VAL 1$: GO SUB 900: GO SUB 2250
: GO SUB 2000: GO SUB 3000: GO T
O 520
9920 PRINT AT 4,7;"PULSE DE NUEV
O": BEEP 1,1: GO TO 505
9950 FOR I=1 TO tim: NEXT I: RET
URN
```

DATOS ANEXOS

NOMBRES DE CALLES

```
C$(1)=SALIDA
C$(2)=RONDA DE VALENCIA
C$(3)=PLAZA DE LAVAPIES
C$(4)=IMPUESTOS: 20000 PTS
C$(5)=ESTA. DE GOYA
C$(6)=REINA VICTORIA
C$(7)=SUERTE
C$(8)=CUATRO CAMINOS
C$(9)=BRAVO MURILLO
C$(10)=CARCEL
C$(11)=GLORIETA DE BILBAO
C$(12)=ALBERTO AGUILERA
C$(13)=FUENCARRAL
C$(14)=ESTA. DE DELICIAS
C$(15)=AVENIDA DE FELIPE II
C$(16)=CAJA DE COMUNIDAD
C$(17)=VELAZQUEZ
C$(18)=SERRANO
C$(19)=PARQUE GRATUITO
C$(20)=AVDA. DE AMERICA
C$(21)=SUERTE
C$(22)=MARIA MOLINA
C$(23)=CEA BERMUDEZ
C$(24)=ESTA. DEL MEDIODIA
C$(25)=PLAZA DE OPERA
C$(26)=CAJA DE COMUNIDAD
C$(27)=BAILEN
C$(28)=PLAZA DE CIBELES
C$(29)=A LA CARCEL
C$(30)=ALCALA
C$(31)=PUERTA DEL SOL
C$(32)=CAJA DE COMUNIDAD
C$(33)=GRAN VIA
C$(34)=ESTA. DEL MEDIODIA
C$(35)=SUERTE
C$(36)=CASTELLANA
C$(37)=IMPUESTOS: 10000 PTS
C$(38)=PASEO DEL PRADO
```

PROGRAMA DE INTRODUCCION

```
9960 FOR I=1TO 38
9970 INPUT "CALLE: ";C$(I)
9975 PRINT "C$(";I;")=";C$(I)
9980 NEXT I
```

```
Q(1,1)=0
Q(1,2)=0
Q(1,3)=0
Q(1,4)=0
Q(1,5)=0
Q(1,6)=0
Q(1,7)=0
Q(1,8)=0
Q(2,1)=200
Q(2,2)=1000
Q(2,3)=3000
Q(2,4)=9000
Q(2,5)=16000
Q(2,6)=25000
Q(2,7)=5000
Q(2,8)=3000
Q(3,1)=400
Q(3,2)=2000
Q(3,3)=6000
Q(3,4)=18000
Q(3,5)=32000
Q(3,6)=45000
Q(3,7)=5000
Q(3,8)=3000
Q(4,1)=0
Q(4,2)=0
Q(4,3)=0
Q(4,4)=0
Q(4,5)=0
Q(4,6)=0
Q(4,7)=0
Q(4,8)=0
Q(5,1)=2500
Q(5,2)=5000
Q(5,3)=10000
Q(5,4)=20000
Q(5,5)=10000
Q(5,6)=0
Q(5,7)=0
Q(5,8)=0
Q(6,1)=600
Q(6,2)=3000
Q(6,3)=9000
Q(6,4)=27000
Q(6,5)=40000
Q(6,6)=55000
Q(6,7)=5000
Q(6,8)=5000
Q(7,1)=-1
Q(7,2)=0
Q(7,3)=0
Q(7,4)=0
Q(7,5)=0
Q(7,6)=0
Q(7,7)=0
Q(7,8)=0
Q(8,1)=600
Q(8,2)=3000
Q(8,3)=9000
Q(8,4)=27000
Q(8,5)=40000
Q(8,6)=55000
Q(8,7)=5000
Q(8,8)=5000
Q(9,1)=800
Q(9,2)=4000
Q(9,3)=10000
Q(9,4)=30000
Q(9,5)=45000
Q(9,6)=60000
Q(9,7)=5000
Q(9,8)=6000
Q(10,1)=0
Q(10,2)=0
Q(10,3)=0
Q(10,4)=0
Q(10,5)=0
Q(10,6)=0
Q(10,7)=0
Q(10,8)=0
Q(11,1)=1000
Q(11,2)=5000
Q(11,3)=15000
Q(11,4)=45000
Q(11,5)=62500
Q(11,6)=75000
Q(11,7)=10000
Q(11,8)=7000
Q(12,1)=1000
Q(12,2)=5000
Q(12,3)=15000
Q(12,4)=45000
Q(12,5)=62500
Q(12,6)=75000
Q(12,7)=10000
Q(12,8)=7000
Q(13,1)=1200
Q(13,2)=6000
Q(13,3)=18000
Q(13,4)=50000
Q(13,5)=70000
Q(13,6)=90000
Q(13,7)=10000
Q(13,8)=8000
Q(14,1)=2500
Q(14,2)=5000
Q(14,3)=10000
Q(14,4)=20000
Q(14,5)=10000
Q(14,6)=0
Q(14,7)=0
Q(14,8)=0
Q(15,1)=1400
Q(15,2)=7000
Q(15,3)=20000
Q(15,4)=55000
Q(15,5)=75000
Q(15,6)=10000
Q(15,7)=9000
Q(15,8)=9000
Q(16,1)=-1
Q(16,2)=0
Q(16,3)=0
Q(16,4)=0
Q(16,5)=0
Q(16,6)=0
Q(16,7)=0
Q(16,8)=0
Q(17,1)=1400
Q(17,2)=7000
Q(17,3)=2000
Q(17,4)=5500
Q(17,5)=75000
Q(17,6)=95000
Q(17,7)=10000
Q(17,8)=9000
Q(18,1)=1600
Q(18,2)=8000
Q(18,3)=22000
Q(18,4)=60000
Q(18,5)=80000
Q(18,6)=100000
Q(18,7)=10000
Q(18,8)=10000
Q(19,1)=0
Q(19,2)=0
Q(19,3)=0
Q(19,4)=0
Q(19,5)=0
Q(19,6)=0
Q(19,7)=0
Q(19,8)=0
Q(20,1)=1800
Q(20,2)=9000
Q(20,3)=25000
Q(20,4)=70000
Q(20,5)=87500
Q(20,6)=105000
Q(20,7)=15000
Q(20,8)=11000
Q(21,1)=-1
Q(21,2)=0
Q(21,3)=0
Q(21,4)=0
Q(21,5)=0
Q(21,6)=0
Q(21,7)=0
Q(21,8)=0
Q(22,1)=1800
Q(22,2)=9000
Q(22,3)=25000
Q(22,4)=70000
Q(22,5)=87500
Q(22,6)=105000
Q(22,7)=15000
Q(22,8)=11000
Q(23,1)=2000
Q(23,2)=10000
Q(23,3)=30000
Q(23,4)=75000
Q(23,5)=92500
Q(23,6)=110000
Q(23,7)=15000
Q(23,8)=12000
Q(24,1)=2500
Q(24,2)=5000
Q(24,3)=10000
Q(24,4)=20000
Q(24,5)=10000
Q(24,6)=0
Q(24,7)=0
Q(24,8)=0
Q(25,1)=2200
Q(25,2)=11000
Q(25,3)=33000
Q(25,4)=80000
Q(25,5)=97500
Q(25,6)=115000
Q(25,7)=15000
Q(25,8)=13000
Q(26,1)=-1
Q(26,2)=0
Q(26,3)=0
Q(26,4)=0
Q(26,5)=0
Q(26,6)=0
Q(26,7)=0
Q(26,8)=0
Q(27,1)=2200
Q(27,2)=11000
Q(27,3)=33000
Q(27,4)=80000
Q(27,5)=97500
Q(27,6)=115000
Q(27,7)=15000
Q(27,8)=13000
Q(28,1)=2200
Q(28,2)=12000
Q(28,3)=36000
Q(28,4)=85000
Q(28,5)=102500
Q(28,6)=120000
Q(28,7)=15000
Q(28,8)=14000
Q(29,1)=0
Q(29,2)=0
Q(29,3)=0
Q(29,4)=0
Q(29,5)=0
Q(29,6)=0
Q(29,7)=0
Q(29,8)=0
Q(30,1)=2600
Q(30,2)=13000
Q(30,3)=39000
Q(30,4)=90000
Q(30,5)=110000
Q(30,6)=127500
Q(30,7)=20000
Q(30,8)=15000
Q(31,1)=2600
Q(31,2)=13000
Q(31,3)=39000
```



```
Q(31,4)=90000
Q(31,5)=110000
Q(31,6)=127000
Q(31,7)=20000
Q(31,8)=15000
Q(32,1)=-1
Q(32,2)=0
Q(32,3)=0
Q(32,4)=0
Q(32,5)=0
Q(32,6)=0
Q(32,7)=0
Q(32,8)=0
Q(33,1)=28000
Q(33,2)=15000
Q(33,3)=45000
Q(33,4)=100000
Q(33,5)=120000
Q(33,6)=140000
Q(33,7)=20000
Q(33,8)=16000
Q(34,1)=2500
Q(34,2)=5000
Q(34,3)=10000
Q(34,4)=20000
Q(34,5)=10000
Q(34,6)=0
Q(34,7)=0
Q(34,8)=0
Q(35,1)=-1
Q(35,2)=0
Q(35,3)=0
Q(35,4)=0
Q(35,5)=0
Q(35,6)=0
Q(35,7)=0
Q(35,8)=0
Q(36,1)=3500
Q(36,2)=17500
Q(36,3)=50000
Q(36,4)=110000
Q(36,5)=30000
Q(36,6)=150000
Q(36,7)=20000
Q(36,8)=17500
Q(37,1)=0
Q(37,2)=0
Q(37,3)=0
Q(37,4)=0
Q(37,5)=0
Q(37,6)=0
Q(37,7)=0
Q(37,8)=0
Q(38,1)=5000
Q(38,2)=20000
Q(38,3)=60000
Q(38,4)=140000
Q(38,5)=170000
Q(38,6)=200000
Q(38,7)=20000
Q(38,8)=20000
```

PROGRAMA DE INTRODUCCION

```
9960>FOR I=1TO 38
9965 FOR J=1 TO 8
9970 INPUT "PRECIO DE LA
     CALLE: ";Q(I,J)
9975 LPRINT "Q(";I;",";J;")=";
     Q(I,J)
9977 NEXT J
9980 NEXT I
```

TARJETAS DE SUERTE:

```
U$(1,1)=Error de la banca a su
U$(1,2)=favor. Reciba 20000 Pts.
U$(2,1)=Reciba su interes sobre
U$(2,2)=el Prestamo: 2500 Pts.
U$(3,1)=Reciba su renta anual:
U$(3,2)=10000 Pts.
U$(4,1)=En su aniversario reciba
U$(4,2)=de cada jugador 1000 Pts
U$(5,1)=Coloquese en la casilla
U$(5,2)=numero uno.
U$(6,1)=La venta de su stock
U$(6,2)=le Produce 5000 Pts
U$(4,2)=de cada jugador 1000 Pts
U$(5,1)=Coloquese en la casilla
U$(5,2)=numero uno.
U$(6,1)=La venta de su stock
U$(6,2)=le Produce 5000 Pts
U$(7,1)=Premio de belleza
U$(7,2)=reciba 10000 Pts
U$(8,1)=Contribuciones le
U$(8,2)=devuelven 2000 Pts
U$(9,1)=Situese en la casilla
U$(9,2)=numero dos.
U$(10,1)=Pague la factura del
U$(10,2)=medico: 5000 Pts
U$(11,1)=Vaya a la carcel
U$(11,2)=
U$(12,1)=Pague al hosPital
U$(12,2)=10000 Pts.
U$(13,1)=Cobra una herencia
U$(13,2)=de 10000 Pts
U$(14,1)=Pague su Poliza de
U$(14,2)=seguro: 5000 Pts
U$(15,1)=Pague 4000 Pts Por casa
U$(15,2)=y 11000 Por hotel.
U$(16,1)=Multa Por embriaguez:
U$(16,2)=Pague 2000 Pts.
U$(17,1)=Ha ganado el Premio de
U$(17,2)=honradez: 10000 Pts.
U$(18,1)=Vete a la casilla
U$(18,2)=numero 1.
U$(19,1)=La banca arroja un
U$(19,2)=dividendo de 5000 Pts.
U$(20,1)=Vaya a la casilla
U$(20,2)=numero 28.
U$(21,1)=Pague Por gastos
U$(21,2)=escolares 15000 Pts.
U$(22,1)=Pague 2500 Por casa
U$(22,2)=y 10000 Por hotel.
U$(23,1)=Vaya a la casilla
U$(23,2)=numero 27.
U$(24,1)=Vaya a la casilla
U$(24,2)=numero 11.
U$(25,1)=Vaya a la casilla
U$(25,2)=numero 8
U$(26,1)=Sus Prestamos Producen
U$(26,2)=15000 Pts.
U$(27,1)=Vaya a la carcel
U$(27,2)=
U$(28,1)=
U$(28,2)=
```

U$(29,1)=PROGRAMA DE INTRODUCCION

```
9960>FOR I=1TO 27
9965 FOR J=1 TO 2
9970 INPUT "SUERTE:";U$(I,J)
9975 LPRINT "U$(";I;",";J;")=";
     U$(I,J)
9977 NEXT J
9980 NEXT I
```

LONGITUD DE LAS TARJETAS

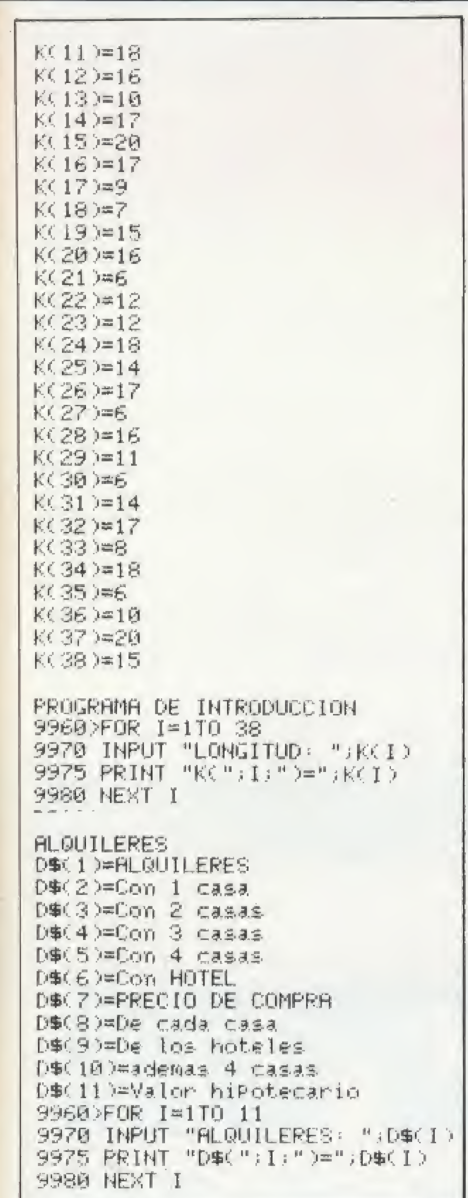
```
V(1,1)=23
V(1,2)=24
V(2,1)=24
V(2,2)=23
V(3,1)=23
V(3,2)=10
V(4,1)=24
V(4,2)=24
V(5,1)=23
V(5,2)=10
V(6,1)=20
V(6,2)=19
V(7,1)=17
V(7,2)=16
V(8,1)=17
V(8,2)=18
V(9,1)=21
V(9,2)=11
V(10,1)=20
V(10,2)=16
V(11,1)=16
V(11,2)=0
V(12,1)=17
V(12,2)=10
V(13,1)=18
V(13,2)=12
V(14,1)=18
V(14,2)=16
V(15,1)=23
V(15,2)=17
V(16,1)=21
V(16,2)=15
V(17,1)=22
V(17,2)=20
V(18,1)=18
V(18,2)=9
V(19,1)=18
V(19,2)=20
V(20,1)=17
V(20,2)=10
V(21,1)=16
V(21,2)=20
V(22,1)=19
V(22,2)=18
V(23,1)=17
V(23,2)=10
V(24,1)=17
V(24,2)=10
V(25,1)=17
V(25,2)=9
V(26,1)=22
V(26,2)=10
V(27,1)=16
V(27,2)=0
```

PROGRAMA DE INTRODUCCION

```
9960>FOR I=1TO 27
9965 FOR J=1 TO 2
9970 INPUT "LONGITUD: ";V(I,J)
9975 LPRINT "V(";I;",";J;")=";
     V(I,J)
9977 NEXT J
9980 NEXT I
```

LONGITUD DE LAS CALLES

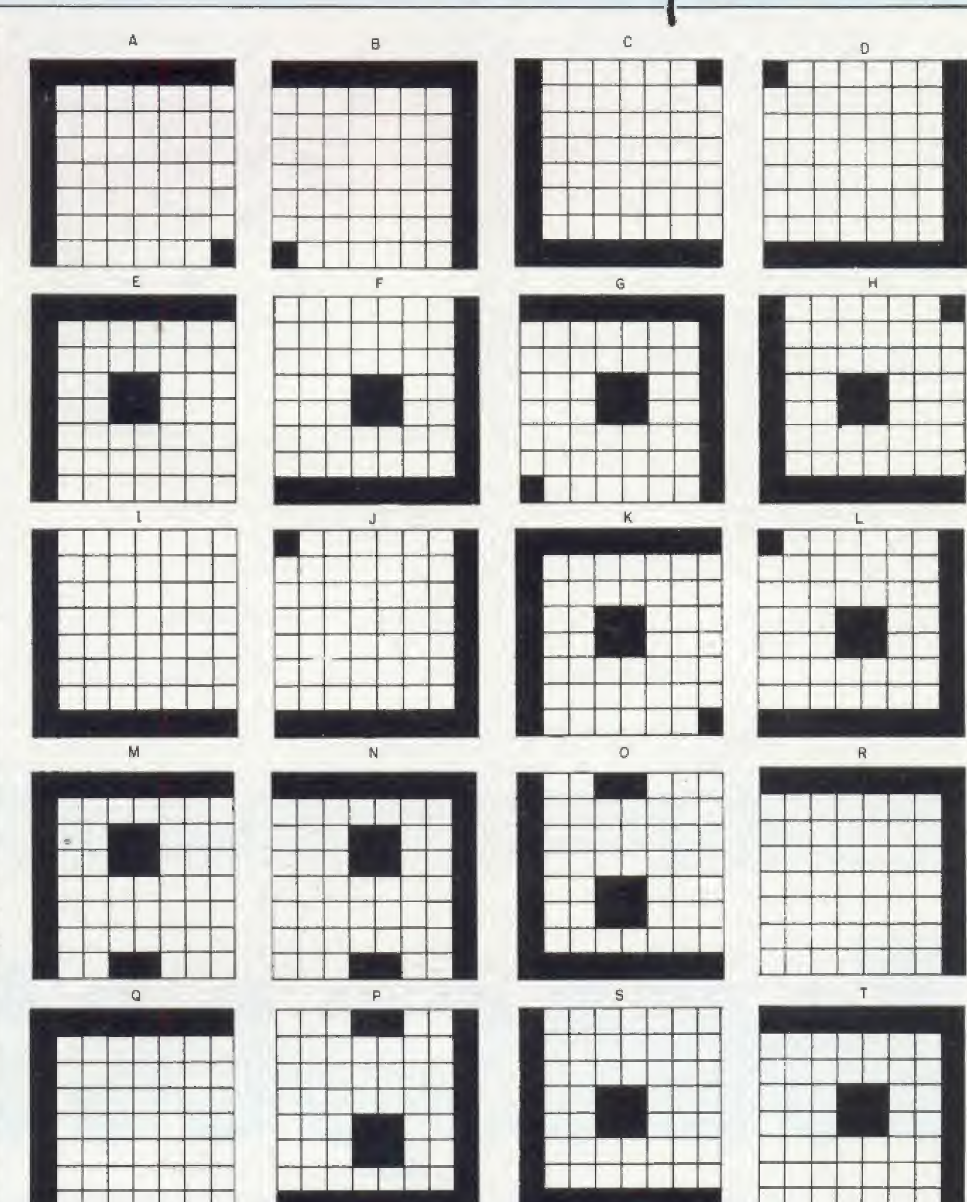
```
K(1)=6
K(2)=17
K(3)=17
K(4)=20
K(5)=13
K(6)=14
K(7)=6
K(8)=14
K(9)=13
K(10)=6
```



```
K(11)=18
K(12)=16
K(13)=10
K(14)=17
K(15)=20
K(16)=17
K(17)=9
K(18)=7
K(19)=15
K(20)=16
K(21)=6
K(22)=12
K(23)=12
K(24)=18
K(25)=14
K(26)=17
K(27)=6
K(28)=16
K(29)=11
K(30)=6
K(31)=14
K(32)=17
K(33)=8
K(34)=18
K(35)=6
K(36)=10
K(37)=20
K(38)=15

PROGRAMA DE INTRODUCCION
9960>FOR I=1TO 38
9970 INPUT "LONGITUD: ";K(I)
9975 PRINT "K(";I;")=";K(I)
9980 NEXT I

ALQUILERES
D$(1)=ALQUILERES
D$(2)=Con 1 casa
D$(3)=Con 2 casas
D$(4)=Con 3 casas
D$(5)=Con 4 casas
D$(6)=Con HOTEL
D$(7)=PRECIO DE COMPRA
D$(8)=De cada casa
D$(9)=De los hoteles
D$(10)=ademas 4 casas
D$(11)=Valor hipotecario
9960>FOR I=1TO 11
9970 INPUT "ALQUILERES: ";D$(I)
9975 PRINT "D$(";I;")=";D$(I)
9980 NEXT I
```



GUSANEZ

por Jose C. Tomas



ESTAREMOS en Informat85

Del 16 al 20 de Abril de 1985



Palacio y Ferial · Nivel 7 · Stand 709
Barcelona del 16 al 20 de Abril de 1985

Maquinas

Y como no todo va a ser jugar al monopoly, en el descanso se puede cargar este programa de grandes posibilidades didácticas, titulado con el nombre de "Máquinas" por su autor, al que, por cierto, pudimos conocer en su domicilio del Polígono de Ocharcoaga en Bilbao, aunque las condiciones urbanísticas no son las más propicias como conocerán los "usuarios del Norte".

Un bonito programa con buenos gráficos, ayudado de algunos caracteres gráficos (subrayados).

Miguel Carmona **16K**

```
   5 REM MAQUINAS
  10 REM  1984 Miguel Carmona*
  20 BORDER 6: PAPER 7: INK 9: G
O SUB 3000
  25 LET zx=90
  30 CLS : REM Reinicio
 510 PRINT AT 0,5;"MAQUINAS SIMP
LES"
 515 PRINT : PRINT TAB 3;"-Maqui
nas son dispositivos destinados
a vencer una fuerza llamada ""Re
sistencia"", mediante la accion
de otra, llamada ""Potencia""."
 520 PRINT : PRINT
 530 PRINT TAB 3;"-Las principal
es son:": PRINT
 540 PRINT TAB 5;"-1A Palanca."
 550 PRINT TAB 5;"-2A Polea fija
."
 560 PRINT TAB 5;"-3A Polea movi
l."
 570 PRINT TAB 5;"-4A Torno."
 580 PRINT TAB 5;"-5A Plano incl
inado."
 590 PRINT TAB 5;"-6A Tornillo."
 600 PRINT TAB 5;"-7A CuNa."
 605 PRINT TAB 5;"-8A PARAR."
 610 PAUSE zx: PRINT AT 20,0;"Pu
lse el nA de la maquina que  de
see"
 620 PAUSE 0: IF INKEY$<"1" OR I
NKEY$>"8" THEN  GO TO 620
 630 IF INKEY$="8" THEN  CLEAR :
 STOP
 640 RESTORE 1000+(100*(CODE INK
EY$-48))
 700 REM Maquinas (cada una)
 710 CLS : IF INKEY$<>"" THEN  G
O TO 710
 720 READ a$,a
 725 IF a=0 THEN  GO TO 880
 730 FOR s=1 TO a: PRINT AT 0,5;
a$
 740 READ b: IF b=0 THEN  GO TO
765
 745 FOR u=1 TO b: READ m,n
 750 READ b$: PRINT AT m,n;b$
 760 NEXT u: READ c,d
 770 IF c=0 THEN  GO TO 820
 780 FOR u=1 TO c
 790 READ co,x,y: PLOT  INK co,x
,y
 800 READ x,y,z: DRAW  INK co;x,
y,z
 810 NEXT u
 820 IF d=0 THEN  GO TO 860
 830 FOR u=1 TO d
 840 READ co,x,y,z: CIRCLE  INK
co;x,y,z
 850 NEXT u
 860 PAUSE zx: PRINT AT 20,0;"Pu
lse una tecla": PAUSE 0
 870 CLS : NEXT s
 880 PRINT AT 0,5;a$: READ c$
 885 PRINT AT 6,0;"Ley de : ";a$
 890 LET p=((32-LEN c$)/2)-2
 900 FOR s=1 TO LEN c$+4: PRINT
AT 9,p+s-1;"-";AT 11,p+s-1;"-":
NEXT s: PRINT AT 10,p;"!";AT 10,
p+LEN c$+3;"!";AT 10,p+2;c$
 910 PAUSE zx: PRINT AT 20,0;"Pu
lse una tecla": PAUSE 0
 920 CLS : READ e
 925 IF e=0 THEN  GO TO 1090
 928 DIM d$(e,32): DIM f$(e,32):
 DIM i(e)
 930 PRINT AT 0,5;a$: FOR s=1 TO
 e: READ d$(s)
 940 PRINT TAB 2;"-";s;"A ";d$(s
)
 950 NEXT s
 960 PAUSE zx: PRINT AT 20,0;"Pu
lse el nA del dato que desea obt
ener"
 970 LET t$=INKEY$: IF CODE t$<4
9 OR CODE t$>e+48 THEN  GO TO 97
0
 980 FOR u=1 TO e: READ f$(u)
 990 NEXT u: LET t=CODE t$-48
1000 CLS : PRINT AT 0,5;a$
1010 PRINT : PRINT "INTRODUCCION
 DE DATOS"
1020 FOR u=1 TO e
1030 IF u=t THEN  GO TO 1050
1040 INPUT (d$(u));" ? ";i(u)
1050 NEXT u
1060 LET r=VAL f$(t)
1070 CLS : PRINT AT 0,5;a$: PRIN
T AT 3,5;c$
1075 FOR z=1 TO 32: IF d$(t,z)="
 " AND d$(t,z+1)=" " AND d$(t,z+
2)=" " THEN  LET z=z-1: GO TO 10
80
1077 NEXT z
1080 PRINT AT 7,0;d$(t);AT 7,z;"
=";r
1090 PAUSE zx: PRINT AT 20,0;"Pu
lse una tecla": PAUSE 0: GO TO 3
0
1095:
1097 REM Palanca
1100 DATA "PALANCA",2
1105 DATA 4,3,3,"-Palanca es una
 maquina constituida por una bar
ra rigida que se puede girar alr
ededor de un punto .",10,23,"p",
12,5,"R",16,1,"-Existen 3 tipos
de palancas :"
1110 DATA 10,1 ,0,52,84,120,0,0
,0,76,84,-8,-8,0 ,0,76,84,8,-8,0
 ,0,68,76,16,0,0 ,0,172,84,0,12,
0 ,0,172,84,4,4,0 ,0,172,84,-4,4
,0 ,0,52,84,0,-12,0 ,0,52,84,-4,
-4,0 ,0,52,84,4,-4,0 ,0,52,100,1
4
1115 DATA 6,3,2,"-1) PALANCA 1A
genero:",3,1,"Es aquella en la
que el punto de apoyo esta situa
do  entre la potencia y la resis
tencia.",8,2,"-2) PALANCA 2A gen
ero:",9,1,"Es aquella en la que
```



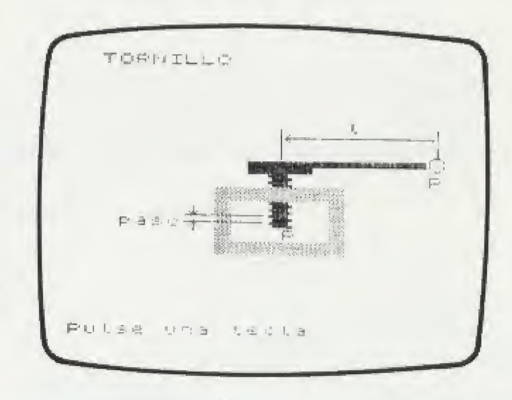
Miguel Carmona en su rincón de trabajo.

(41)

```
 la resistencia actua entre el pu
 nto de apoyo y la potencia.",14,
 2,"-3) PALANCA 3A genero:",15,1,
 "Es aquella en la que la potenci
 a esta situada entre el punto de
  apoyo y la resistencia."
 1120 DATA 0,0,"P*bp=R*br",4
 1125 DATA "Potencia(P).","Brazo
 potencia(bp).","Resistencia(R)."
 ,"Brazo resistencia(br)."
 1130 DATA "(i(3)*i(4))/i(2)","(i
 (3)*i(4))/i(1)","(i(1)*i(2))/i(4
 )","(i(1)*i(2))/i(3)"
 1140:
 1150 REM Polea fija
 1200 DATA "POLEA FIJA",2
 1205 DATA 4,3,2,"Es una maquina
 constituida por un disco que pue
 de girar sobre un eje fijo y tie
 ne en su periferia un canal que
 aloja una cuerda."
 1207 DATA 17,12,"R",16,16,"P",9,
 11,"      "
 1210 DATA 10,2 ,0,110,80,0,0,0 ,
 0,100,80,0,-40,0 ,0,120,80,0,-40
 ,0 ,0,95,40,10,0,0 ,0,95,40,0,-1
 0,0 ,0,95,30,10,0,0 ,0,105,40,0,
 -10,0
 1215 DATA 0,110,80,0,20,0 ,0,120
 ,40,4,4,0 ,0,120,40,-4,4,0 ,0,11
 0,80,10 ,0,110,80,2
 1220 DATA 1,3,2,"La ventaja de
 la  polea fija radica en el hech
 o de que es mas comodo ejercer u
 na fuerza hacia abajo que hacia
 arriba, considerando que puede a
 provecharse como potencia el pro
 pio peso de la persona que manip
 ula la maquina.",0,0
 1225 DATA "P=R",2
 1230 DATA "Potencia(F).","Resist
 encia(R).","i(2)","i(1)"
 1240:
 1250 REM Polea movil
 1300.DATA "POLEA MOVIL",2
 1305 DATA 4,2,2,"Es una maquina
 analoga a la polea fija, con la
 unica diferencia de que el eje d
 e la polea puede moverse en sent
 ido ascendente o descendente."
 1307 DATA 13,20,"P",20,17,"R",13
 ,15,"    "
 1310 DATA 9,2 ,0,130,70,0,-40,0
 ,0,150,70,0,-40,0 ,0,150,70,-4,-
 4,0 ,0,150,70,4,-4,0 ,0,140,30,0
 ,-13,0
 1313 DATA 0,132,17,16,0,0 ,0,132
 ,17,0,-9,0 ,0,132,8,16,0,0 ,0,14
 8,17,0,-9,0 ,0,140,30,10 ,0,140,
 30,2
 1315 DATA 4,2,2,"Fara evitar la
 incomodidad que supone el que la
  potencia sea una fuerza ascende
 nte, en general se emplea una co
 mbinacion de     polea movil
  y polea fija, tal como pued apr
 eciarse en la figura"
 1320 DATA 19,21,"P",20,17,"R",13
 ,15,"      "
 1325 DATA 13,4 ,0,130,70,0,-40,0
  ,0,150,56,0,-26,0 ,0,150,70,-4,
 -4,0 ,0,150,70,4,-4,0 ,0,140,30,
 0,-13,0
 1327 DATA 0,164,20,0,35,0 ,0,164
 ,20,4,4,0 ,0,164,20,-4,4,0 ,0,15
 7,56,0,8,0
 1330 DATA 0,132,17,16,0,0 ,0,132
 ,17,0,-9,0 ,0,132,8,16,0,0 ,0,14
 8,17,0,-9,0 ,0,140,30,10 ,0,140,
 30,2 ,0,157,56,7 ,0,157,56,2
 1335 DATA "P=R/2",2
 1340 DATA "Potencia(P).","Resist
 encia(R).","i(2)/2","i(1)*2"
 1350:
 1360 REM Torno
 1400 DATA "TORNO",1
 1405 DATA 7,2,2,"Torno es una ma
 quina constituida por un cilindr
 o que lleva arrollada una cuerda
  y que puede girar sobre su eje
 mediante la accion de una manive
 la."
 1410 DATA 8,19,"F",18,13,"R",11,
 22,"r2",13,22,"r1",17,12," ",18,
 12," "
 1415 DATA 29,0 ,0,80,70,90,0,0 ,
 0,80,71,70,0,0 ,0,85,80,0,-20,0
 ,0,85,80,55,0,0 ,0,85,60,55,0,0
 ,0,140,80,0,-20,0.
 1420 DATA 0,150,70,0,30,0 ,0,150
 ,100,20,0,0 ,0,150,99,10,0,0 ,0,
 169,98,0,-26,0 ,0,143,60,26,0,0
 1425 DATA 0,169,98,-2,-2,0 ,0,16
 9,98,2,-2,0 ,0,169,72,2,2,0 ,0,1
 69,72,-2,2,0 ,0,169,68,0,-6,0 ,0
 ,169,68,-2,-2,0 ,0,169,68,2,-2,0
  ,0,169,62,2,2,0 ,0,169,62,-2,2,
 0
 1430 DATA 0,100,80,0,-40,0 ,0,10
 3,80,0,-20,0 ,0,106,80,0,-20,0 ,
 0,109,80,0,-20,0 ,0,112,80,0,-20
 ,0 ,0,115,80,0,-20,0 ,0,118,80,0
 ,-20,0 ,0,121,80,0,-20,0 ,0,124,
 80,0,-20,0
 1440 DATA "F/R = r1/r2",4
 1445 DATA "Potencia(P).","Resist
 encia(R).","radio del cilindro(r
 1).","radio de la manivela(r2)."
 1450 DATA "(i(3)*i(2))/i(4)","(i
 (1)*i(4))/i(3)","(i(1)*i(4))/i(2
 )","(i(2)*i(3))/i(1)"
 1460:
 1470 REM Flano inclinado
 1500 DATA "FLANO INCLINADO",1
 1505 DATA 5 ,2,2,"Flano inclinad
 o es una superficie plana que fo
 rma un angulo con el suelo y se
 utiliza para elevar cuerpos."
 1510 DATA 11,20,"F",15,18,"R",14
 ,24,"h",12,15,"1"
 1515 DATA 24,1 ,0,80,40,90,0,0 ,
 0,170,40,0,35,0 ,0,80,40,90,35,0

 1520 DATA 0,171,40,0,55,0 ,0,80,
 39,110,0,0 ,0,171,75,19,0,0 ,0,8
 0,41,0,19,0
 1525 DATA 0,82,55,86,35,0 ,0,82,
 55,4,0,0 ,0,82,55,3,3,0 ,0,86,59
 ,0,0,0 ,0,168,90,-4,0,0 ,0,168,9
 0,-3,-4,0
 1530 DATA 0,185,41,0,32,0 ,0,185
 ,41,3,3,0 ,0,185,41,-3,3,0 ,0,18
 5,73,-2,-2,0 ,0,185,73,2,-2,0
 1535 DATA 0,141,71,15,6,0 ,0,140
 ,69,0,-20,0 ,0,156,77,-3,0,0 ,0,
 156,77,-2,-3,0 ,0,140,49,3,3,0 ,
 0,140,49,-3,3,0
 1537 DATA 0,140,70,5
 1540 DATA "F/R = h/1",4
 1545 DATA "Potencia(P).","Resist
 encia(R).","altura(h).","longitu
 d(1)."
 1550 DATA "(i(3)*i(2))/i(4)","(i
 (1)*i(4))/i(3)","(i(1)*i(4))/i(2
 )","(i(2)*i(3))/i(1)"
 1600 DATA "TORNILLO",2
 1605 DATA 1,2,0,"El tornillo es
 una maquina cons-tituida por una
  ""cabeza"", sobre la que se apl
 ica la potencia, y una parte cil
 indrica, sobre  la que se  ha  t
 allado un surco de seccion  tria
 ngular o cuadrangu-lar, que se d
 enomina ""rosca"" y  que describ
 e una espiral."
 1610 DATA 0,0
 1620 DATA 13 ,9,25,"F",5,20,"1",
 12,6,"paso"
 1625 DATA 10,12,"GGGGGGG",11,12
 ,"G        G",12,12,"G        G",13,
 12,"G        G",14,12,"GGGGGGG"
 1630 DATA 8,14,"     [redacted]",9,1
 5,"CD",11.15,"CD",12,15,"CD"
 1635 DATA 13.16,"R"
 1637 DATA 14,1 ,0,128,130,0,-17,
 0 ,0,205,130,0,-15,0 ,0,130,125,
 73,0,0
 1640 DATA 0,130,125,2,2,0 ,0,130
 ,125,2,-2,0 ,0,203,125,-2,2,0 ,0
 ,203,125,-2,-2,0
 1642 DATA 0,80,79,39,0,0 ,0,80,7
 5,39,0,0 ,0,85,70,0,14,0 ,0,85,7
 9,-2,2,0 ,0,85,79,2,2,0 ,0,85,75
 ,-2,-2,0 ,0,85,75,2,-2,0 ,0,204,
 110,4
 1645 DATA "P/R = p/(2*H*1)",4
 1650 DATA "Potencia(P).","Resist
 encia(R).","longitud llave(1).",
 "paso(p)."
 1655 DATA "(i(2)*i(4))/(2*PI*i(3
 ))","(i(1)*2*PI*i(3))/i(4)","(i(
 2)*i(4))/(i(1)*2*PI)","(i(1)*2*P
 I*i(3))/i(2)"
 1690:
 1695 REM cuNa
 1700 DATA "CUMA",2
 1705 DATA 9 ,2,1,"La cuNa es un
 prisma tiangular construido con
 un material duro y tenaz, que se
  emplea para rom-per o separar c
 uerpos."
 1710 DATA 12,9,"P",12,13,"R",10,
 4,"A",9,8,"O",10,13,"B",15,9,"P"
 ,19,8,"C",12,16,"Q"
 1715 DATA 19,0 ,0,40,90,60,0,0 ,
 0,40,90,30,-70,0 ,0,100,90,-30,-
 70,0
 1717 DATA 0,70,90,0,-30,0 ,0,70,
 60,2,2,0 ,0,70,60,-2,2,0
 1720 DATA 0,70,58,0,-5,0 ,0,70,5
 0,0,-5,0 ,0,70,43,0,-5,0 ,0,70,3
 6,0,-5,0 ,0,70,28,0,-3,0
 1725 DATA 0,70,90,50,-15,0 ,0,12
 0,74,-1,3,0 ,0,120,74,-3,0,0 ,0,
 70,90,-50,-15,0 ,0,20,74,3,0,0 ,
 0,20,74,1,3,0
 1730 DATA 2,121,73,-50,-14,0 ,2,
 20,73,50,-14,0
 1750 DATA 16 ,1,5,"__",2,5,"OF=p
 otencia=P",3,5,"__",4,5,"OQ=resi
 stencia=R",5,5,"__",6,5,"AB=cabe
 za de la cuNa=b",7,5,"__",8,5,"A
 C=lado de la cuNa=1"
 1755 DATA 12,9,"P",12,13,"R",10,
```

```
4,"A",9,8,"O",10,13,"B",15,9,"P"
,19,8,"C",12,16,"Q"
1760 DATA 19,0 ,0,40,90,60,0,0 ,
0,40,90,30,-70,0 ,0,100,90,-30,-
70,0
1765 DATA 0,70,90,0,-30,0 ,0,70,
60,2,2,0 ,0,70,60,-2,2,0
1770 DATA 0,70,58,0,-5,0 ,0,70,5
0,0,-5,0 ,0,70,43,0,-5,0 ,0,70,3
6,0,-5,0 ,0,70,28,0,-3,0
1775 DATA 0,70,90,50,-15,0 ,0,12
0,74,-1,3,0 ,0,120,74,-3,0,0 ,0,
70,90,-50,-15,0 ,0,20,74,3,0,0 ,
0,20,74,1,3,0
1780 DATA 2,121,73,-50,-14,0 ,2,
20,73,50,-14,0
1785 DATA " P/R = b/l ",4
1790 DATA "Potencia(P).","Resist
encia(R).","cabeza de la cuNa(b)
.","lado de la cuNa(l)."
1795 DATA "(i(2)*i(3))/i(4)","(i
(1)*i(4))/i(3)","(i(1)*i(4))/i(2
)","(i(2)*i(3))/i(1)"
2999 STOP
3000 REM graficos
3005 RESTORE 3100
3010 FOR i=0 TO 7
3020 READ a: POKE USR "a"+i,a
3030 READ c: POKE USR "c"+i,c
3035 READ d: POKE USR "d"+i,d
3040 READ n: POKE USR "n"+i,n
3050 NEXT i
3090 REM datas
3100 DATA 0,63,240,124 ,24,63,24
0,0 ,24,15,252,120 ,0,15,252,68
,60,63,240,68 ,0,63,240,68 ,0,15
,252,68 ,0,15,252,0
3200 FOR i=0 TO 7 STEP 2: POKE U
SR "g"+i,BIN 10101010: POKE USR
"g"+i+1,BIN 01010101: NEXT i
3300 RESTORE 3350: FOR i=0 TO 7
3310 READ h: POKE USR "h"+i,h
3320 NEXT i
3350 DATA 0,0,126,36,36,36,38,0
3390 RESTORE 3420
3400 FOR i=0 TO 7: READ m: POKE
USR "m"+i,m: NEXT i
3420 DATA 56,66,98,82,74,70,66,0
3500 RESTORE 3570
3510 FOR i=0 TO 7
3520 READ a,b,c
3530 POKE USR "o"+i,a: POKE USR
"p"+i,b: POKE USR "q"+i,c
3540 NEXT i
3570 DATA 1,1,1 ,61,125,61 ,66,6
6,66 ,66,124,64 ,126,66,64 ,66,6
6,66 ,66,124,60 ,0,0,0
4000 RETURN
```



Archivo

Si su archivo manual se ha convertido en un caos, es hora de pasar a un buen programa de manejo de datos. Pero antes de rebuscar en la tienda especializada, ¿por qué no echar una ojeada a éste que le ofrecemos? Es muy sencillo y, sobre todo, muy rígido, ya que está preparado para un hipotético "almacén" con veinte productos y la siguiente información por producto: descripción del artículo, precio venta público, coste, stock, ventas año anterior, y descripción punto de pedido. Reducidas dimensiones, pero buen ejemplo de cómo tratar la información en tablas.

El menú incluye las opciones de ALTAS para nuevos datos, BAJAS para anulación de artículos, MODIFICACION para cambiar la información, CONSULTA con la que poder visualizar los artículos, GRABAR para guardar las tablas de datos, y finalmente METER DATOS ANTIGUOS para introducir datos previos.

En su versión original, el programa incluye pantalla de carga (programa 1), pasando posteriormente al programa principal (programa 2). Por carecer de interés, omitimos los bytes de la pantalla de carga.

Ricardo López *16K*

Programa 1

```
  10 LOAD "rot."SCREEN$ : BORDER
 0: LOAD "ARCHIVO"
9999 LOAD "rot.": BORDER 0: LOAD
 "ARCHIVO"
```

Programa 2

```
   1 CLS : BORDER 7: DIM I$(1200
): LET I$(1)="*": DIM A$(18)
   2 DIM B$(60): LET B$(60)="*"
   3 DIM P$(7): DIM C$(7): LET Z
$="  "
   6 DIM S$(8): DIM V$(8): DIM T
$(8)
   7 POKE 23609,50: POKE 23658,8
  10 CLS : PRINT AT 4,11; " M E
N U  ": PRINT AT 5,11;" ======="
  30 PRINT AT 8,9;"1.-   ALTAS"
  40 PRINT AT 10,9;"2.-   BAJAS"
  50 PRINT AT 12,9;"3.-   MODIFI
CACION"
  60 PRINT AT 14,9;"4.-   CONSUL
TA"
  65 PRINT AT 16,9;"5.-   GRABAR
"
  67 PRINT AT 18,9;"6.-   METER
DATOS "
  68 PRINT AT 19,9;"     ANTIGUOS"
  69 PLOT 0,0: DRAW 255,0: DRAW
0,165: DRAW -255,0: DRAW 0,-165:
 PLOT 3,3: DRAW 249,0: DRAW 0,12
3: DRAW -249,0: DRAW 0,-123
  80 IF INKEY$="1" THEN  GO TO 2
00
  90 IF INKEY$="2".THEN  GO TO 3
00
 100 IF INKEY$="3" THEN  GO TO 4
00
 110 IF INKEY$="4" THEN  GO TO 5
00
 112 IF INKEY$="5" THEN  GO TO 2
000
 113 IF INKEY$="6" THEN  GO TO 3
000
 115 GO TO 80
 200 CLS : PRINT AT 9,12;"ALTAS"
: PRINT AT 10,12;"=====": PAUSE
35: CLS
 225 PRINT AT 10,3;"
                 CUANDO YA NO Q
UIERA METER MAS  DATOS INTRODUZ
CA UNA N EN EL   ARTICULO": PLO
T 0,55: DRAW 250,0: DRAW 0,40: D
RAW -250,0: DRAW 0,-40
 230 PRINT AT 5,8;"INTRODUZCA :"
: INPUT AT 5,12;"ARTICULO :";A$:
 IF A$="N                 " THEN
 GO TO 10
 231 INPUT AT 5,12;"P.V.P ";P$
 232 INPUT AT 5,12;"COSTE ";C$
 234 INPUT AT 5,12;"STOCK EN ALM
AC. ";S$
 235 INPUT AT 5,12;"VENTAS ANO A
NT. ";V$
 236 INPUT AT 5,12;"PUNTO DE PED
IDO ";T$
 238 LET L$=A$+Z$+P$+C$+S$+V$+T$
 239 LET L=LEN L$
 240 FOR I=0 TO 1200 STEP 60
 250 LET A=I: IF I=0 THEN  LET A
=I+1
 254 IF I=1200 THEN  GO TO 265
 255 IF I$(A TO A+2)="*  " THEN
 GO TO 260
 256 GO TO 265
 260 LET L$=L$+B$(1+L TO 60): LE
T L$(60)="*": LET I$(A+1 TO I+60
)=L$: LET I$(I+60)="*": LET L$="
": GO TO 279
 265 IF I=1200 THEN  CLS : PRINT
 "NO LE QUEDA SITIO PARA METER D
ATOS . PARA MENU PULSE CUALQUIER
 TECLA": PAUSE 0: GO TO 10
 270 NEXT I
 290 GO TO 230
 300 CLS : PRINT AT 9,12;"BAJAS"
 310 PRINT AT 10,12;"====="
 320 PRINT AT 12,0;"____________
____________________"
 330 PRINT AT 15,2;"INTRODUCE NU
MERO DE ARTICULO A  DAR DE BAJA.
 PARA IR A MENU :   INTRODUCE  0
 "
 335 PRINT
 340 INPUT TAB (12);"NUMERO DE A
RT. ";X
```



```
 350 IF X=0 THEN  GO TO 10
 355 IF X=1 THEN  LET I$(X*60-58
TO X*60-1)*"": GO TO 340
 360 LET I$(X*60-59 TO X*60-1)="
"
 370 GO TO 340
 400 CLS : PRINT AT 9,10;"MODIFI
CACION": PLOT 0,90: DRAW 255,0:
INPUT ;"NUMERO DE ARTICULO A MOD
IFICAR  (PARA SALIR 0) : ";V: IF
V=0 THEN  GO TO 10
 405 IF V=1 THEN  LET K$=I$(V*60,
-58 TO V*60-1)
 407 IF V=1 THEN  GO TO 410
 408 LET K$=I$(V*60-59 TO V*60-1
)
 410 CLS : PRINT AT 4,3;"QUE DES
EA MODIFICAR ": PLOT 15,130: DRA
W 230,0
 415 PRINT AT 8,6;"1.-ARTICULO";
AT 9,6;"2.-P.V.P";AT 10,6;"3.-CO
STE";AT 11,6;"4.-STOCK EN ALMACE
N";AT 12,6;"5.-VENTAS ANO ANTERI
OR";AT 13,6;"6.-PUNTO DE PEDIDO"
: PLOT 15,55: DRAW 230,0: DRAW 0
,65: DRAW -230,0: DRAW 0,-65
 418 PRINT AT 17,3;"Mete numero
correspondiente,     si no deseas
 modificar mas      pulsa 0 y EN
TER ": INPUT W
 421 IF W=0 THEN  GO TO 400
 423 IF W=1 THEN  LET N=1: LET W
=18
 424 IF W=2 THEN  LET N=21: LET
W=27
 425 IF W=3 THEN  LET N=28: LET
W=34
 426 IF W=4 THEN  LET N=35: LET
W=42
 427 IF W=5 THEN  LET N=43: LET
W=50
 428 IF W=6 THEN  LET N=51: LET
W=58
 430 LET M$=K$(N TO W)
 435 CLS : PRINT AT 19,8;"- DATO
 ANTIGUO ";M$
 436 INPUT AT 2,8;"- DATO NUEVO
";T$
 440 IF V=1 THEN  LET I$((60*V-5
9)+N TO (60*V-59)+W)=T$
 445 IF V=1 THEN  GO TO 410
 448 LET I$((60*V-60)+N TO (60*V
-60)+W)=T$
 450 GO TO 410
 500 CLS : PRINT AT 2,12;"CONSUL
TA": PRINT AT 3,12;"========": P
RINT AT 5,0;"__________________
____________"
 505 PRINT
 510 FOR I=0 TO 1200 STEP 60
 515 LET A=I: IF I=0 THEN  LET A
=1
 525 IF A=1200 THEN  PRINT : INP
UT ;" PARA"; FLASH 1;" AMPLIAR I
NFORMACION"; FLASH 0;" DE UN  AR
T. PULSA SU NUMERO Y  ENTER ; SI
NO,PULSE  0  "; FLASH 0;Y: GO T
O 5000
 540 IF I$(A TO I+2)="    " THEN
GO TO 620
 545 LET L$=I$(A TO A+60)
 560 FOR P=1 TO 60
 570 IF L$(P TO P+1)<>"   " THEN
GO TO 610
 575 IF I$(A TO A+2)="*   " THEN
GO TO 620
 580 PRINT "Art: ";I$(A TO A+P-1
);TAB (25);"NUM. ";(1+I/60): GO
TO 620
```

```
      ART. 1        MICROS
 ------------------------------
   P.V.P.                45000
   COSTE                 23000
   STOCK EN ALMAC.       200000
   VENTA ANO ANT.        45000
   PUNTO DE PEDIDO       DEALERS
 ------------------------------
   PARA VER MAS ARTICULOS PULSE
   NUM. CORRESPOND.,SI NO  0
```

```
 610 NEXT P
 620 NEXT I
2000 CLS : PRINT AT 11,5;"
                         PUEDE GRAB
AR CUANDO LO DESEE.    PULSE G PA
RA GRABAR O S PARA     SALIR AL M
ENU": PLOT 0,85: DRAW 250,0: DRA
W 0,-35: DRAW -250,0: DRAW 0,35
2020 IF INKEY$="G" OR INKEY$="g"
 THEN  GO TO 2050
2025 IF INKEY$="S" OR INKEY$="s"
 THEN  GO TO 10
2030 GO TO 2020
2050 CLS : SAVE "DATOS" DATA I$(
)
2060 CLS : FLASH 1: PRINT AT 10,
5;"YA ESTA GRABADO": FLASH 0: PR
INT AT 12,2;"PARA VERIFICAR CAMB
IE CLAVIJAS Y PULSE V"
2070 IF INKEY$="V" OR INKEY$="v"
 THEN  GO TO 2085
2080 GO TO 2070
2085 CLS : PRINT AT 12,2;"SI POR
 CUALQUIER MOTIVO,DESEA    SALIR
DEL VERIFY PULSE BREAK      Y CUAN
DO ESTE PARADO ,NUNCA     RUN ,U
TILIZE EL GOTO 10": PLOT 5,40: D
RAW 250,0: DRAW 0,47: DRAW -250,
0: DRAW 0,-47
2090 VERIFY "DATOS" DATA I$()
2100 GO TO 10
3000 CLS : PRINT AT 11,5;"
                         PUEDE MET
ER CUANDO LO DESEE.    PULSE M P
ARA METER O S PARA     SALIR AL
MENU": PLOT 0,85: DRAW 250,0: DR
AW 0,-35: DRAW -250,0: DRAW 0,35
3020 IF INKEY$="M" OR INKEY$="m"
 THEN  GO TO 3050
3025 IF INKEY$="S" OR INKEY$="s"
 THEN  GO TO 3040
3030 GO TO 3020
3040 GO TO 10
3050 CLS : PRINT AT 11,5;"INRODU
ZCA CLAVIJAS EN            EAR
Y PULSE PLAY.": LOAD "DATOS" DAT
A I$()
3060 CLS : FLASH 1: PRINT AT 10,
8;" YA ESTA METIDO ": FLASH 0: P
RINT AT 16,3;"PARA MENU PULSE CU
ALQUIER        TECLA": PAUSE 0
3070 GO TO 10
5000 CLS : IF Y=0 THEN  GO TO 10
5002 IF Y<>1 THEN  GO TO 5008
5004 LET M=Y*60-58: LET K=M+18:
GO TO 5010
5008 LET M=Y*60-59: LET K=M+19
5010 PRINT AT 5,3;" ART. ";Y,I$(
M TO K)
5020 PRINT AT 7,0;"______________
___________________"
5030 PRINT AT 10,3;"P.V.P.";TAB
(23);I$(M+20 TO M+26);AT 11,3;"C
OSTE";TAB (23);I$(M+27 TO M+33)
5040 PRINT AT 12,3;"STOCK EN ALM
AC.";TAB (23);I$(M+34 TO M+41):
PRINT AT 13,3;"VENTA ANO ANT.";T
AB (23);I$(M+42 TO M+49): PRINT
AT 14,3;"PUNTO DE PEDIDO";TAB (2
3);I$(M+50 TO M+57): PLOT 10,48:
 DRAW 245,0: DRAW 0,55: DRAW -24
5,0: DRAW 0,-55
5050 PRINT AT 18,2;"PARA VER MAS
 ARTICULOS PULSE     NUM. CORRESP
OND.,SI NO  0 "
5052 IF INKEY$="" THEN  GO TO 50
52
5055 IF  INKEY$<>"0" THEN  GO TO
 5070
5065 GO TO 10
5070 CLS : INPUT AT 15,5;"NRO. D
E ART. ";Y: GO TO 5000
```

Un poco de Código Máquina

Muchas calculadoras de bolsillo incorporan la función factorial (marcado normalmente como x!). El factorial de un número x, se define de la siguiente forma: x! (leído como x factorial), es igual al número de las distintas combinaciones que se pueden realizar con x objetos. Por ejemplo, si tiene 4 libros en una estantería, puede colocarlos de veinticuatro formas diferentes (4! = 24).

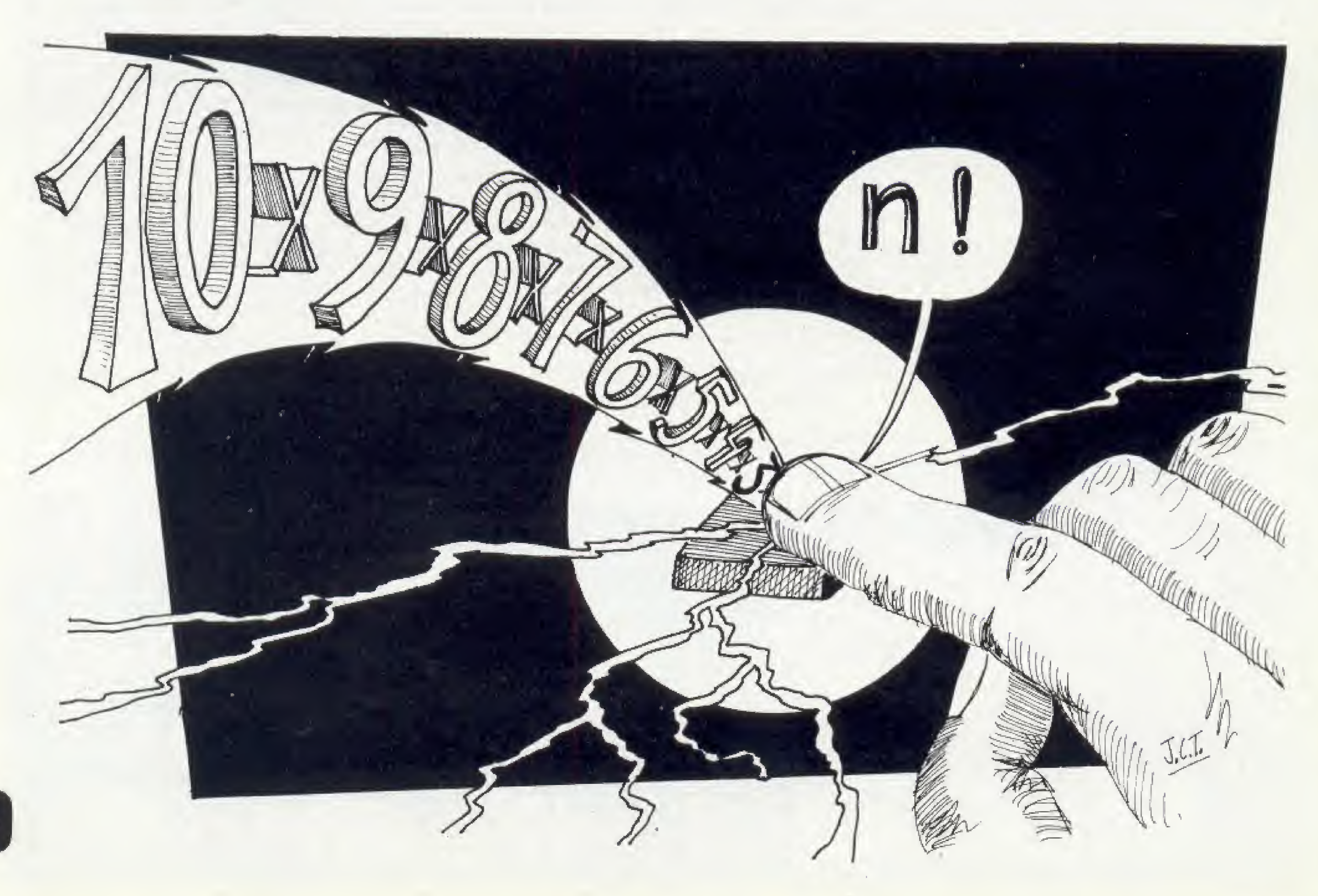
Son muchas las aplicaciones de los factoriales en el campo de la matemática pura. Uno de los principales usos es en el cálculo de series infinitas para ampliar distintas funciones.

Aunque incorporados en muchas calculadoras, es una función que rara vez encontrará en un microordenador. Con la pequeña rutina que le ofrecemos a continuación, su Spectrum ya no tendrá que envidiar a la más completa calculadora. Aunque la función factorial no se utiliza con tanta frecuencia como otras (seno, raíz cuadrada, etc.), su ventaja estriba en la posibilidad de realizar iteraciones de series infinitas para lograr mejores aproximaciones en los cálculos de las diferentes funciones con que trabaje.

La forma en que se calcula el factorial se puede resumir de la siguiente forma: Se toma un número y se multiplica por ese mismo número menos uno, despues por éste menos uno y así sucesivamente hasta llegar al número uno. Por si las matemáticas no son su fuerte, veamos un par de ejemplos:

$4! = 4 \times 3 \times 2 \times 1 = 24$
$6! = 6 \times 5 \times 4 \times 3 \times 2 \times 1 = 270$

El factorial de 1 es 1. Pero el factorial de 0 también es 1. Sólo hay



una forma de ordenar ningún objeto, al igual que sólo hay una forma de ordenar un objeto. El método es sencillo en términos de programación, y sólo requiere una línea en BASIC:

```
LET FACT = 1 : FOR F = 1 TO X : LET FACT = FACT × F : NEXT F
```

que funciona bastante bien siempre que no desee obtener el factorial de un número alto, digamos por encima de 5. La obtención del factorial de 30 ya puede ser "excesivo" para el BASIC de Sinclair. Si lo coloca como una subrutina habrá de incorporar la instrucción RETURN lo que ocasionará un mayor retardo en la respuesta. Respuesta que, en términos de velocidad, viene siempre de la mano salvadora del código máquina. Para mayor precisión, de los 142 insignificantes bytes que requiere esta rutina.

¿Dónde almacenar esta rutina? Se puede elegir una zona de memoria protegida o situarla dentro de la esfera de acción del processor interrupts o interrupciones del procesador con lo que la rutina sería llamada 50 veces por segundo. Por el momento, para no complicar el tema, y dada la reducida longitud de la misma, puede colocarla en una dirección alta (la instrucción permite introducir la dirección donde se desea colocar la rutina).

Para utilizar la nueva función factorial incorporada a su Spectrum (una vez cargado y ejecutado este programa), introduzca en la variable "x" el valor del que desea obtener el factorial (línea 310). Después sólo tiene que llamar a la subrutina, de acuerdo a la dirección donde la ubicó:

```
RANDOMIZE USR comienzo dirección
```

obteniéndose el valor factorial de "x" en la variable "fact".

Como podrá comprobar, no sólo funciona sino que se obtiene el resultado de forma casi instantánea. Sólo hay que tener en cuenta inicializar previamente la variable "fact" antes de llamar a la subrutina. De otra forma se produciría el mensaje "Variable not found" como ocurriría con cualquier otra variable en BASIC. Recuerde:

```
LET fact = 0
```

Finalmente le damos dos expresiones matemáticas típicas que requieren este tipo de cálculo. La fórmula de MacLaurin se utiliza para trabajar con funciones simples, y la expresión binomial normalmente sirve para trabajar con funciones del tipo $(1 + x)^n$.

Maclaurin's Expansion:

$$f(x) = f(0) + f'(0)x + \frac{f''(0)x^2}{2!} + \dots + \frac{f^{(r)}(0)x^r}{r!} + \dots$$

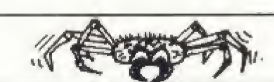
$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + \frac{(-1)^n x^{2n+1}}{(2n+1)!} + \dots$$

$$e^x = + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots \frac{x^r}{r!} + \dots$$

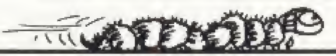
General Binomial Expansion:

$$(1 + x)^n = 1 + nx + \frac{n(n-1)x^2}{2!} + \frac{n(n-1)(n-2)x^3}{3!} + \dots + \frac{n(n-1)(n-2)\dots\dots(n-r+1)x^r}{r!} + \dots$$

Y en el próximo número...



SPECTRUM PARLANCHIN, rutina que hace hablar al ordenador.



CONTROL STOCKS, para controlar la situación.



AULA INFORMATICA CON SPECTRUM, y con el Logo.

QL-DIVERSION. Análisis del Easel y primer QL-listado.

Preguntas y respuestas

P Me gustaría que me explicaseis cómo habéis dibujado la portada de vuestro número 3 con el GIZMO, y especialmente cómo habéis logrado el color marrón e intercalarlo con el negro.

Por otro lado, os quisiera preguntar a qué se refiere el manual cuando dice que si se alteran algunas variables del sistema mediante POKE éste queda alterado produciendo efectos perjudiciales. ¿Se daña irreparablemente con esto al Spectrum? ¿Se daña sólo el programa que está en memoria? ¿Se daña sólo hasta que se desconecta de la red?

J. M. Martínez
Cádiz

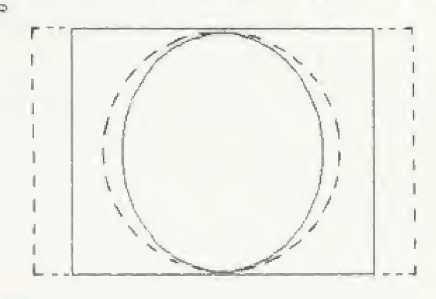
R *Varias personas nos han preguntado cómo hemos logrado el color marrón en esa portada, y también otros tonos poco habituales. La respuesta es: ¡No lo sabemos! Son los famosos "duendes de la imprenta" los que hacen que los colores no respondan muchas veces a la fotografía original. En este caso el efecto fue muy conseguido, otras veces no tanto. En fin, en la pantalla original el color era rojo.*

Las variables del sistema a que se refiere el manual son esenciales para el buen funcionamiento de la máquina. Cuando se dice que su alteración daña al sistema se refiere a que pueden "colgar" (alterar el funcionamiento) el sistema operativo del Spectrum. En cualquier caso, su efecto no durará más que hasta que hagamos RANDOMIZE USR 0 o desconectemos la corriente. Lo que sí pueden causar es la pérdida del programa o los datos que estuviéramos ejecutando. Así que, si queréis hacer experimentos, salvad antes en cassette *vuestro programa. No se puede causar ningún daño permanente al Spectrum usando el teclado. Sí se puede estropear si conectamos o desconectamos algún periférico con la alimentación conectada.*

P Tengo una impresora Seikosha GP-50S y he observado lo mucho que distorsiona la imagen alargándola. Desearía que me constestaseis si esto es normal o es un fallo propio de mi máquina. Adjunto listado e impresión, en la que marco en trazo discontinuo el dibujo de una circunferencia hecha con un compas, donde se advierte la distorsión producida y las dimensiones del recuadro para guardar las mismas proporciones que el dibujo que aparece en la pantalla de televisión.

Clifford Wait
Cádiz

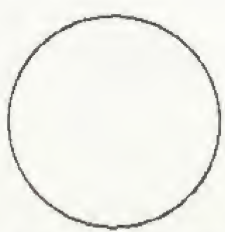
```
  10 CLS : DRAW 255,0: DRAW 0,17
5: DRAW -255,0: DRAW 0,-175: CIR
CLE 127,87,85: PLOT 127,87
  20 LLIST : LPRINT : COPY : STO
P
```



R *Remitimos su pregunta a Dirak, distribuidores de las impresoras Seikosha, quienes pasan a contestarle:*

"Normalmente casi todas las impresoras, poseen un ratio de proporción de los puntos, que provoca un pequeño desfase entre el dibujo de pantalla y el obtenido por COPY en la impresora.

*Nos remitimos al manual de la impresora, sección de características (pág. 8, apartado 1-F) donde dice que el espaciado entre puntos es 1/84 (horizontal) * 1/72 (vertical). Todo esto implica que, para corregir el ratio en el ejemplo que nos muestra, tendrá que realizar el siguiente programa:*



```
 5 LET r=80: LET r1=80*72/84
 7 LET c1=128: LET c2=86
10 FOR x=0 TO 2*PI STEP .01
20 PLOT c1+r*COS x,c2+r1*SIN x
30 NEXT x
```

El corcho

SE VENDE ZX Spectrum 48 K con manuales, accesorios, y los mejores programas comerciales (juegos y utilidades), a estrenar, garantía investrónica con la fecha en blanco: 27.000. Impresora GP50S con 12 rollos de papel, a estrenar, garantía en blanco: 21.500. TU BM 12" ELBE Mod. 1215: 9000. Urge mucho. José Rafael Díaz. Antonio Maura, 36, 2 der. Oviedo. Tlfno. 253795.

INTERESADOS EN FORMAR UN CLUB DE USUARIOS DEL SPECTRUM DE LA PROVINCIA DE SEGOVIA, PARA INTERCAMBIAR IDEAS, PROGRAMAS, ETC. DIRIGIRSE A LUIS MORENO MARINAS. TORIBIO SANZ N.º 3. COCA (SEGOVIA)

VENDO Spectrum 48 K (4 meses de uso, garantía), impresora Seikosha GP100 (profesional), interface Centronics RS232, programas de contabilidad y declaración renta y cinco juegos originales. Dos libros de Spectrum, 15 números de la revista ZX y varios de TODOSPECTRUM, así como las mejores inglesas. Además, póliza de seguro a todo riesgo (robo, incendio, averías, manejo inadecuado, defecto fábrica...). Todo por 89.000 ptas. Precio de coste superior a las 120.000 ptas. Santiago de Biedma. Virgen de los Rosales, 12. Madrid 28023.

VENDO AMPLIACION EXTERNA A 48 Kb DE INDESCOMP, COMO NUEVA, CON SU EMBALAJE ORIGINAL, POR 7.000 PTAS. LLAMAR AL 340 72 28 DE VALENCIA (PREFIJO 96), PREGUNTAR POR JOSEP (NOCHES).

CAMBIO Transceptor 40 canales, 2 antenas (móvil y Balconera), medidor de estacionarias y potencia, cables, libros, revistas por Interface 1, con microdrive o por Impresora Seikosha GP50S o vendo por 21.000 pts. Interesados llamar al 4378285 de Vizcaya, preguntar por Guillermo. Se estudiarán otras ofertas.

Envíanos tu carta para "El corcho", TODOSPECTRUM c/ Bravo Murillo 377 - 5º A 28020 Madrid.

TU SPECTRUM...
MAS POTENTE

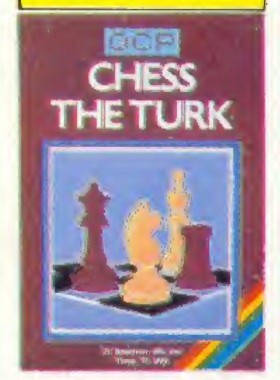
J M PUBLICIDAD

SINCLAIR STORE, te ofrece la colección de cintas OCP.

LO MAS POTENTE PARA SU SPECTRUM

Con ellas podrás obtener un mayor rendimiento de tu ordenador personal.
Si aún no las conoces, solicita una demostración, sin compromiso,
en tu boutique informática SINCLAIR STORE más próxima.
¡Quedarás sorprendido!

PRECIO OFERTA 1.250 PTAS.



- 6 niveles de juego.
- "Al paso" y "Casting".
- Coronación de peones y transformación en 4 piezas diferentes.
- Partidas relámpago.
- Ahorra jornadas y tablas comunes.
- Opción de ayuda.
- Salida de todas las jugadas a impresora.
- Opción de demostración.

PRECIO ~~3.000~~ PTAS. OFERTA 1.750 PTAS.



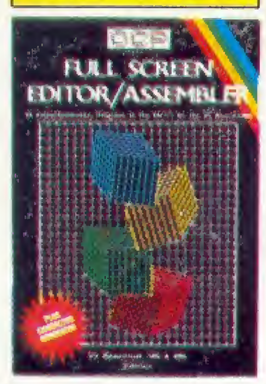
Si tiene interés en averiguar cómo realmente funciona un ordenador, si desea experimentar el poder auténtico de programación, "la herramienta de comprobación de códigos", es la respuesta.

"La herramienta de comprobación de códigos" es un programa de utilización que viene completo con un curso de instrucción, permitiéndole dar entrada, comprobar, mostrar en pantalla y dominar rutinas de códigos de máquina hexadecimales, de manera fácil y rápida.

"La herramienta de comprobación de códigos", está pensada para ayudar al principiante que desea explorar este fascinante tema, o al experto que desea mejorar sus programas de códigos de máquina.

Una rutina de conversión HEXadecimal está incluida en el programa como standard.

PRECIO ~~3.000~~ PTAS. OFERTA 1.750 PTAS.



- Input/editor de texto de pantalla completa de 42 columnas, simplicidad, velocidad y facilidad de manejo.
- Assembly a pantalla, impresora y/o cinta (velocidad de assembly 1.000 líneas/minuto).

- Juego completo instrucciones Z80.
- Constantes Binary, Octal, Hex y ASCII.
- Comprobación amplia de sintaxis y mensajes de error, mostrando código de origen, línea, etc.

- Instrucciones Assembler DEFM, DEFS, DEFW, DEFB, DEFL, EQU, END, ORG.
- Valoración de expresiones sofisticadas: paréntesis colocados dentro de otros, aritmética modular, operadores lógicos, etc.

- Mover, copiar e imprimir en bloque.
- Símbolo/nombre de marca de 6 caracteres.
- Ordenes "stand alone" and "quash assembler" permiten manejo de texto aún en 16k.

- Muestra conjunto de símbolos.
- Opción "wait on error".
- String find and substitute, etc.
- Funcionará también como procesor de palabra simple.
- Podrá ser colocado en memoria para de-bugging con el MACHINE CODE TEST TOOL.

PRECIO ~~3.000~~ PTAS. OFERTA 1.750 PTAS.



Es un programa rápido de aplicación con código de máquina que ofrece a los usuarios del Spectrum un archivo profesional de direcciones, índices, y sistema de recuperación.

ADDRESS MANAGER ha sido diseñado cuidadosamente para proveer al usuario de un programa que es extremadamente amistoso y fácil de usar.

Incluye las siguientes características:

- Tiene capacidad para 400 nombres completos, direcciones y n.° de teléfono en 48 K.
  O más de 1.500 nombres individuales/títulos/referencias.
- Indice múltiple - el índice triple definido por el usuario permite la clasificación exacta de entradas, esencial para un uso más sofisticado.
- Entradas completas mostradas en pantalla - una manera avanzada y muy útil de dar entrada a datos, fácil y rápidamente.
- Respuesta instantánea - este programa es muy rápido.

Su uso incluye:

En casa: archivar y actualizar direcciones, imprime lista de tarjetas de navidad, etc.

En el trabajo: emite listas de direcciones, clasificación de clientes por tipos, áreas, tamaños; a su elección.

En el ocio: administración de miembros de un club, catálogos, y mucho, mucho más.

PRECIO ~~3.000~~ PTAS. OFERTA 1.750 PTAS.



Sus características principales son:

- ENTRADA DOBLE AUTOMATICA - para hasta 255 cuentas separadas, con totales corrientes.
- ANALISIS - por descripciones definidas por el usuario y/o fecha.
- PEDIDOS EN CARTERA - sumados, aplicados y eliminados por fechas o en intervalos regulares.
- SALDOS - todos los saldos de cuentas pueden ser listados en cualquier fecha específica.

- FICHA DE TRANSACCIONES NUEVAS - diferencia entre entradas antiguas y nuevas.
- ORDEN DE PAGINA - pasa a través de todas las cuentas, hacia adelante y hacia atrás.
- CONCILIACION - permite al usuario controlar artículos incorrectos en los extractos.

- PRESENTACION COMPLETA EN PANTALLA - vea lo que ocurre cuando da entrada, rectifica o borra.
- VELOCIDAD - este programa es muy, muy rápido.

PRECIO ~~3.000~~ PTAS. OFERTA 1.750 PTAS.



Sus características principales son:

- Reloj de tiempo real y alarma, con conexión/desconexión e indicaciones impresas.
- 10 teclas de funciones programables, hasta 252 caracteres.
- Mover, copiar, borrar un bloque de líneas, fusionar dos líneas.
- Encontrar y cambiar conjuntos de caracteres.
- Mostrar en pantalla mapa de memoria gráfica - liberando la memoria restante.
- Re-numerar programa básico.
- Trazar - traza la ejecución de un programa BASIC con retraso variable entre cada extracto.
- Variables - enseña todas las variables usadas en un programa y sus valores.
- RAMTOP - cambia RAMTOP sin borrar las variables.
- COMPRESS, REMKILL y PACK, para reducir al mínimo el número de bytes usados en un programa.
- Salida a la impresora - para imprimir texto sobre papel.
- Cambiar caja - cambia la caja superior alpha a la caja inferior y viceversa.

En todas las cintas, existe la "VERSION PLUS 80" (80 columnas, compatibles con interface centronic KEMPSTON). (Al precio especial de 2.000 ptas. cada una).



SOMOS PROFESIONALES

BRAVO MURILLO, 2 (aparc. gratuito en C/. Magallanes, 1). Tel.: 446 62 31
DIEGO DE LEON, 25 (aparc. gratuito en C/. Núñez de Balboa, 114). Tel.: 261 88 01 MADRID